Koyo

Value & Technology

可编程序控制器 **DL05 系列**

用户手册

[第一版]

光洋电子(无锡)有限公司

前言

感谢你购买 DL05 系列 PLC 产品。本手册向你详细介绍了 DL05 系列 PLC 的安装、设置及维护等内容；它也帮助你理解如何在一个系统中连接 DL05 系列 PLC 和其他的设备。在首次使用 DL05 系列 PLC 前，务请先详细阅读本手册。

如果你在阅读本手册或使用 DL05 系列 PLC 时有什么疑难，或你需要另外的信息时，请与本公司或各地办事机构联系，以便尽快得到帮助。

由于产品的改进等原因，有时资料所刊内容会与实际的产品有些差别，请注意！

在没有预先得到本公司书面许可的情况下，此手册的任何一部分都不得以任何方式被拷贝，复制或传播。本公司保留对包括此文件在内的所有信息的专利权。

有关编程的内容，请参见《S 系列编程手册》

有关 PID 调节等功能，参见《DL05 特殊功能/单元技术资料》

手册版本

如果有关此手册的情况你需与我们联系的话，应确认版本号。

题目：DL05 系列 用户手册

刊号：KEW-M4111A

版本	日期	变更说明
原版	**01/07**	**初版**

本手册中所用的提示标记

当你看到左边的“感叹号”图标时，表明紧随其后的段落是一条**警告**信息。此信息可以防止伤害、财产损失，甚至死亡。

左边的“记事本”图标表明紧随其后的段落将是一条特别的注意事项。

目　　录

第 1 章 开始 1

DL05 系列 PLC 1
一般规格 1
供电电源规格 2
I/O 选择 2
输入接线图 2
连接电源线 4
连接编程设备 4

第 2 章　安装 5

DL05 前面板 5
端子台拆卸 5
安装尺寸 6
导轨安装 7
输入/输出回路保险丝 7
源/汇的概念 8
I/O 公共端"COMMON"的概念 8
连接 DC 型 I/O 与晶体管现场设备("SOLID STATE") 9
继电器输出接线方法 12
DC 输入接线方法 14
DC 输出接线方法 15
AC 输入/AC 输出接线方法 16
高速 I/O 接线方法 17
DL05 I/O 电气性能规格 18

第 3 章　　CPU 性能和操作 ..... 错误！未定义书签。

简介 20
DL05 特点 20
CPU 规格 21
CPU 硬件设置 22
状态指示灯 23
运行方式开关 24
系统参数区 24
停电保持区域 24
DL05 功能存储器 25
DL05 系统参数寄存器 26
DL05 特殊继电器 28
DL05 网络系统设置 30

第 4 章　高速计数和脉冲输出功能 39

第一节　简介 39
第二节　模式 10：高速计数 41
第三节　模式 20：A/B 相计数 51
第四节　模式 30：脉冲输出 56
第五节　模式 40：外部中断 74
第六节　模式 50：脉冲捕捉 78
第七节　模式 60：滤波输入 80

第 5 章　系统维护 82

硬件系统的维护 82
诊断 82
CPU 显示 85
通讯问题 85
I/O 点问题的解决 85
电气干扰问题 86
程序调试 86

附录 87

附录一　DL05 指令表 87
附录二　DL05 系列无协议通讯 93

第1章 开始

DL05 系列 PLC

DL05 系列有 8 种型号，外观和 CPU 性能都相同。CPU 指令系统和 DL240 CPU 相同，加上一些机械控制应用特殊指令。所有 DL05 PLC 都有 2 个 RS-232C 通讯口。DC 输入型的前 3 个输入点可以用作高速输入。DC 输出型的前 2 个输出点可以用作脉冲输出。所有 DL05 PLC 都有大容量的程序存储器和丰富的指令、先进的诊断功能。

DL05 系列 PLC 为整体式不可扩展 PLC，但其本体上有一个扩充插槽，可装入一块特殊的扩充单元，例如：模拟量处理单元，日历时钟功能单元等等。

DL05型号	输入类型	输出类型	外部电源	高速输入	脉冲输出
D0-05AR	AC	继电器	95-240VAC	No	No
D0-05DR	DC	继电器	95-240VAC	Yes	No
D0-05AD	AC	DC	95-240VAC	No	Yes
D0-05DD	DC	DC	95-240VAC	Yes	Yes
D0-05AA	AC	AC	95-240VAC	No	No
D0-05DA	DC	AC	95-240VAC	Yes	No
D0-05DR-D	DC	继电器	12-24V DC	Yes	No
D0-05DD-D	DC	DC	12-24V DC	Yes	Yes

一般规格

项目	规格
保存环境温度	-20℃~70℃
使用环境温度	0℃~55℃
环境湿度	5~95%（无凝露）
环境气氛	无腐蚀性气体
耐震动	MIL STD 810C，Method514.2
耐冲击	MIL STD 810C，Method516.2
抗干扰	NEMA（ICS3-304 标准）
标准取得	UL，CE，FCC A 类

供电电源规格

项目	交流电源供电 DL05 单元	直流电源供电 DL05 单元
输入电压范围	110/220VAC（95—240VAC）	12-24VDC（10.8— 26.4VDC）
最大浪涌电流	13A，1ms(95—240VAC) 15A，1ms(240—264VAC)	10A
最大消耗功率	30VA	20W
耐压	1500VAC 1 分钟	
绝缘	>10MΩ 在 500VAC	

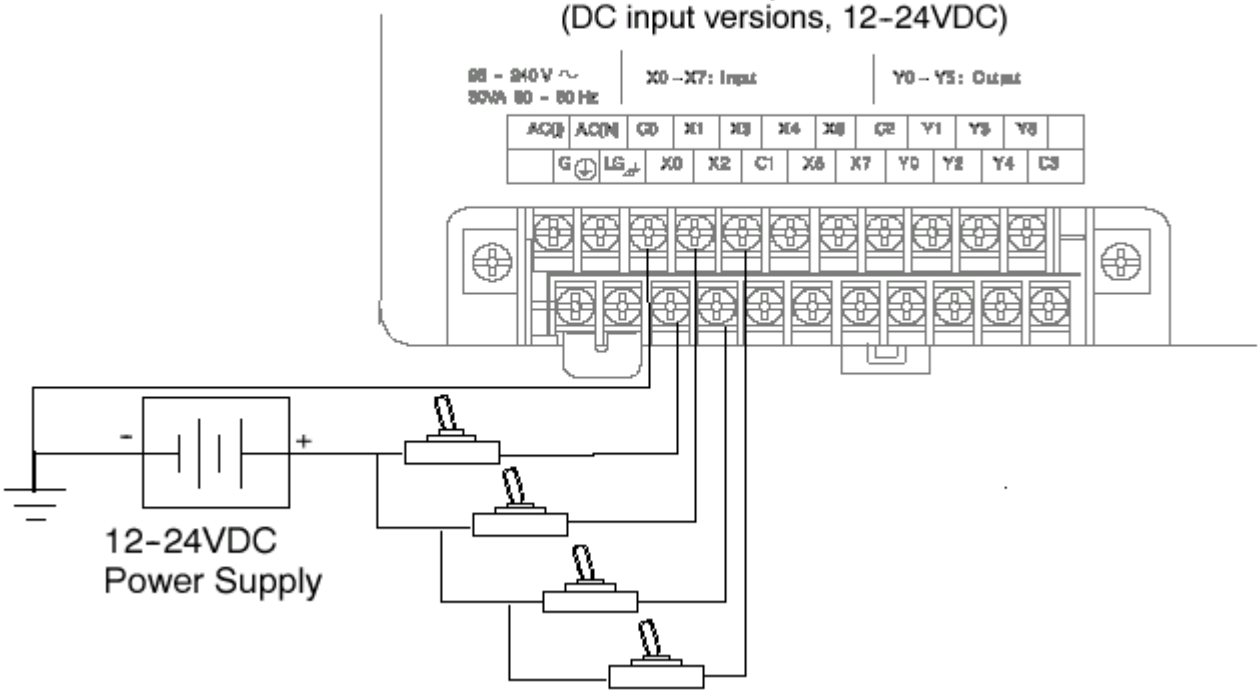
I/O 选择

DL05 型号	输入			输出		
	类型/ 公共端	源/汇	电压范围	类型/ 公共端	源/汇	电压/电流范围
D0-05AR	AC / 2	--	90-120VAC	继电器/2	源或汇	6-27VDC,2A 6-240VAC,2A
D0-05DR	DC / 2	源或汇	12-24VDC	继电器/2	源或汇	6-27VDC,2A 6-240VAC,2A
D0-05AD	AC / 2	--	90-120VAC	DC/1	汇	6-27VDC,0.5A(Y0-2) 6-240VAC,1A(Y3-5)
D0-05DD	DC / 2	源或汇	12-24VDC	DC/1	汇	6-27VDC,0.5A(Y0-2) 6-240VAC,1A(Y3-5)
D0-05AA	AC / 2	--	90-120VAC	AC/2	--	17-240VAC,47-63Hz, 0.5A
D0-05DA	DC / 2	源或汇	12-24VDC	AC/2	--	17-240VAC,47-63Hz, 0.5A
D0-05DR-D	DC / 2	源或汇	12-24VDC	继电器/2	源或汇	6-27VDC,2A 6-240VAC,2A
D0-05DD-D	DC / 2	源或汇	12-24VDC	DC/1	汇	6-27VDC,0.5A(Y0-2) 6-240VAC,1A(Y3-5)

输入接线图

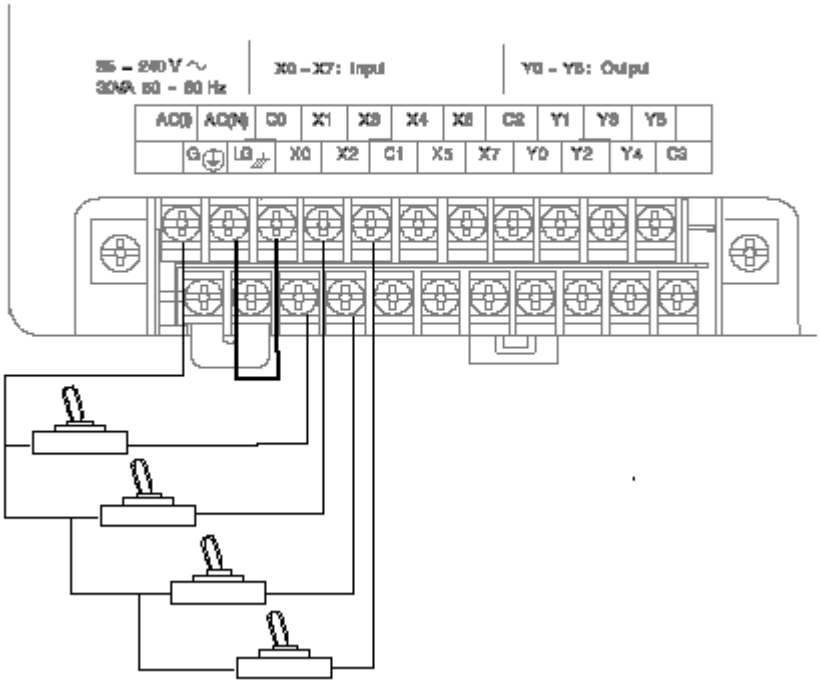
如果是 DC 输入，需要外加 12-24VDC 电源。

D0-05DR, D0-05DD, D0-05DA
D0-05DR-D, D0-05DD-D
(DC input versions, 12-24VDC)



警告：不要用 240VAC 按下图接线，开关量输入只能为 120VAC。

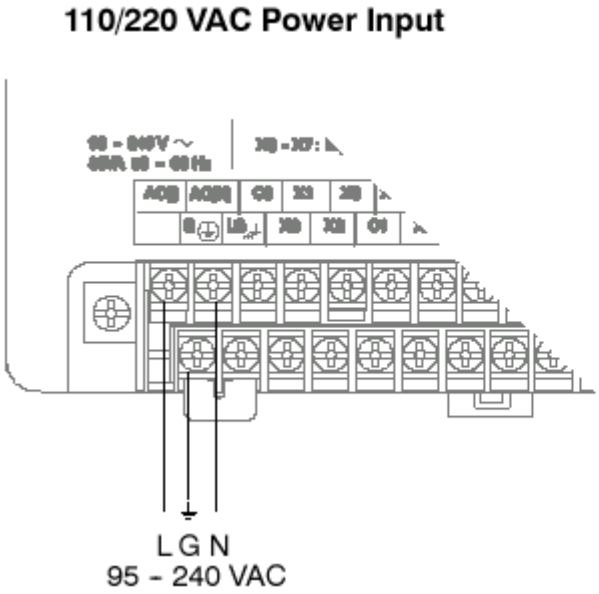
D0-05AR, D0-05AD, D0-05AA
(AC input versions, 120VAC only)



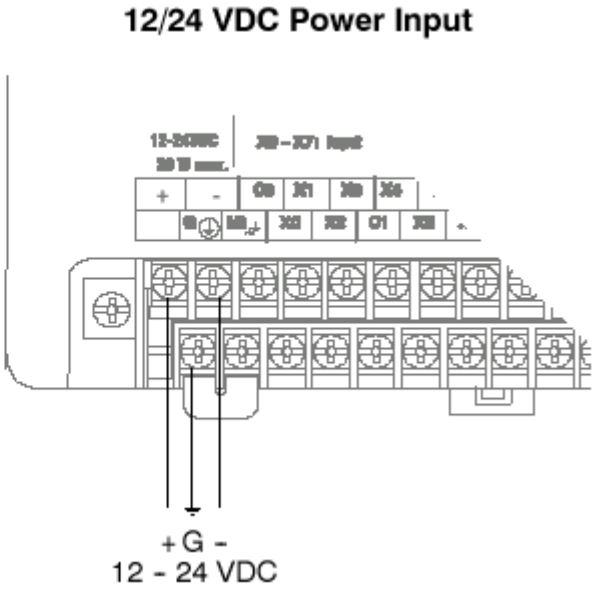
连接电源线

详细的安装和接线在第 2 章还有介绍。

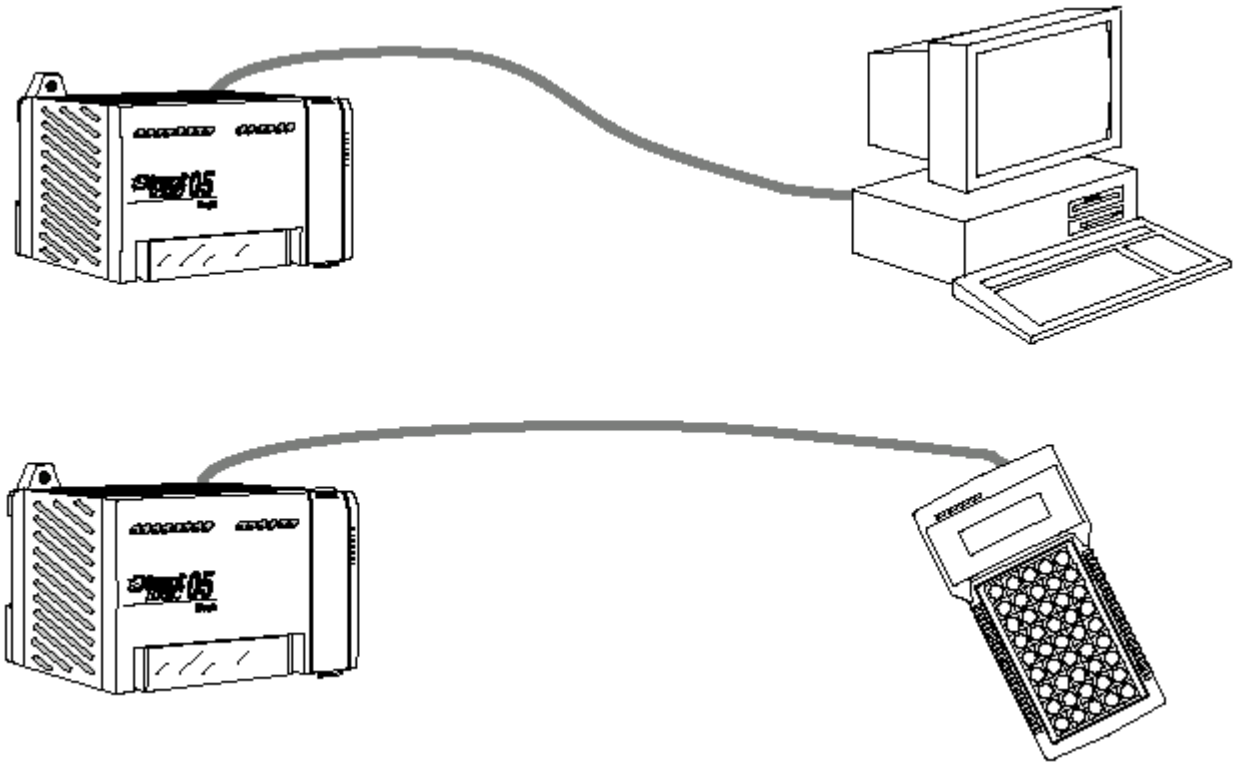
110/220VAC 电源



12/24VDC 电源

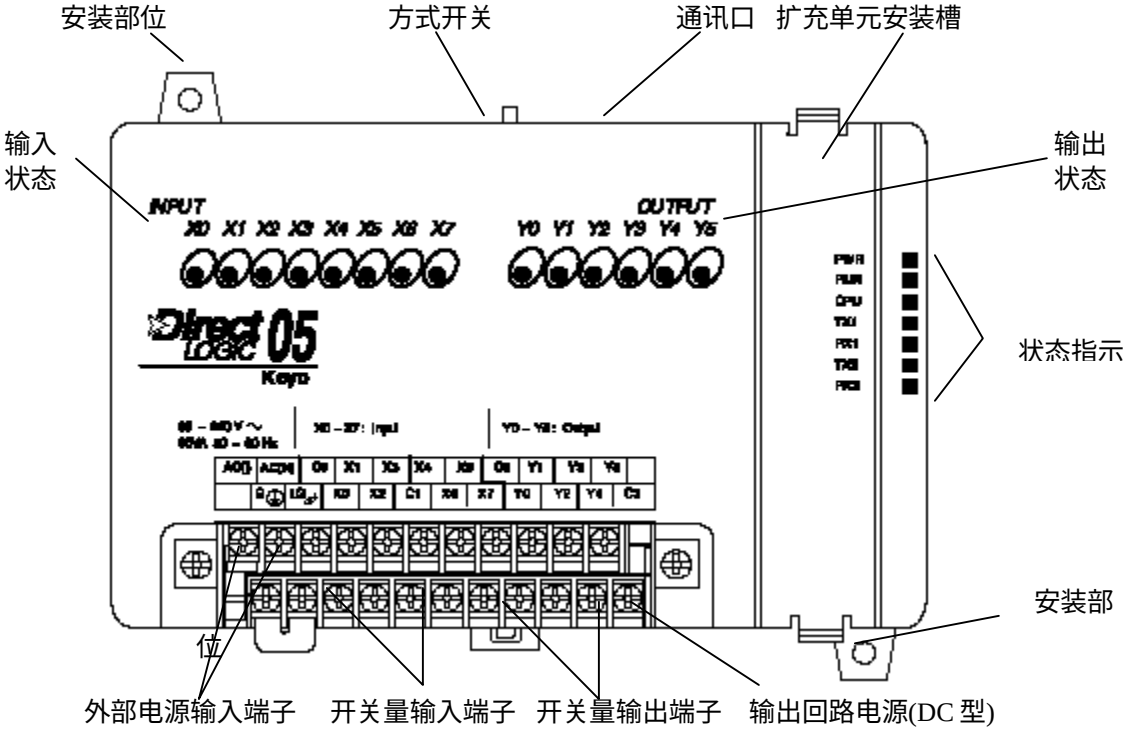


连接编程设备



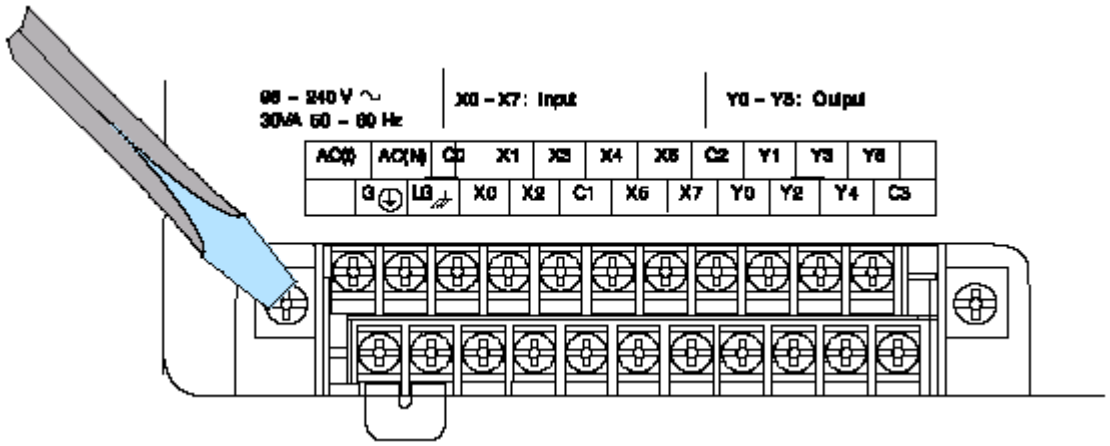
第 2 章 安装

DL05 前面板



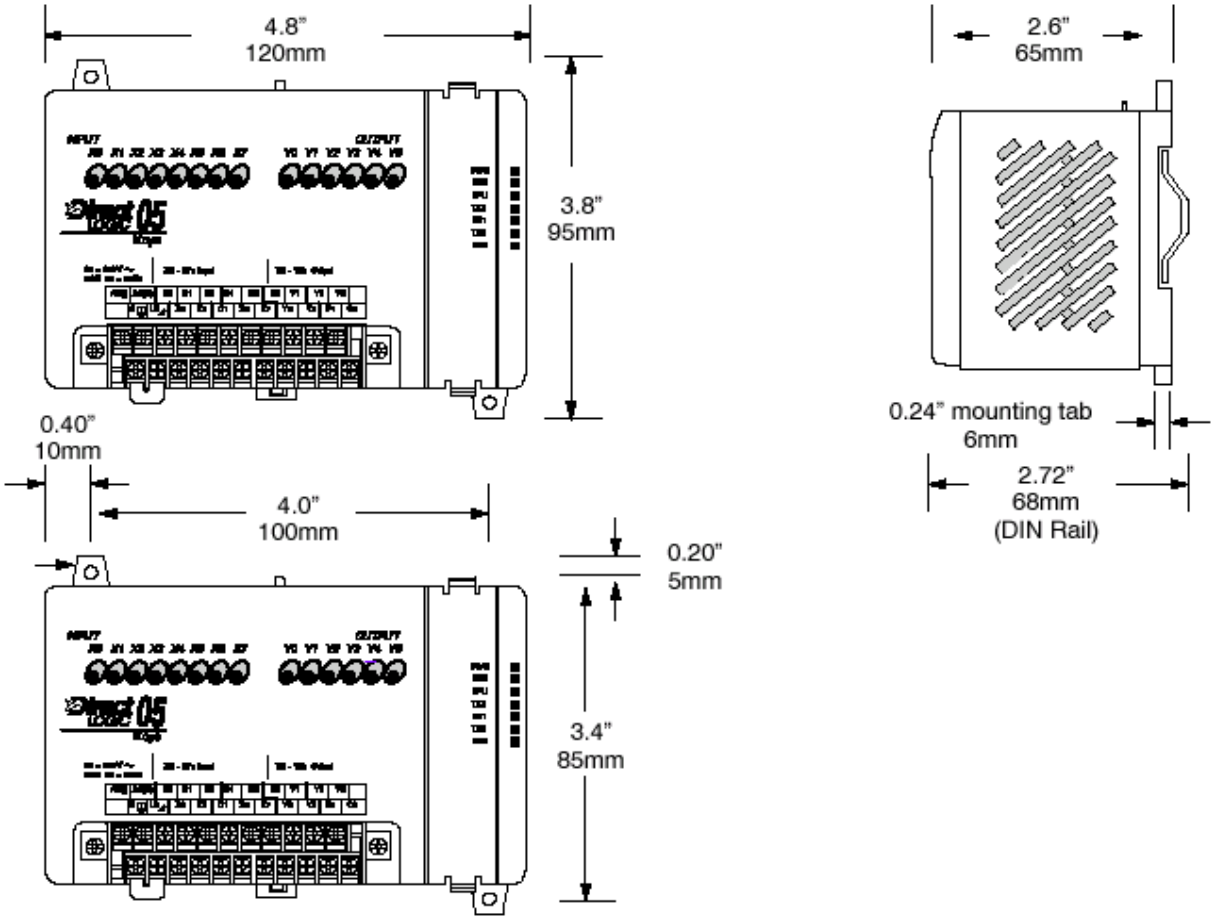
端子台拆卸

所有 DL05 的接线端子都设计在一个可以拆卸的端子台上，该端子台用 2 个小螺丝固定，可以容易拆卸，下图为其拆卸示意图。



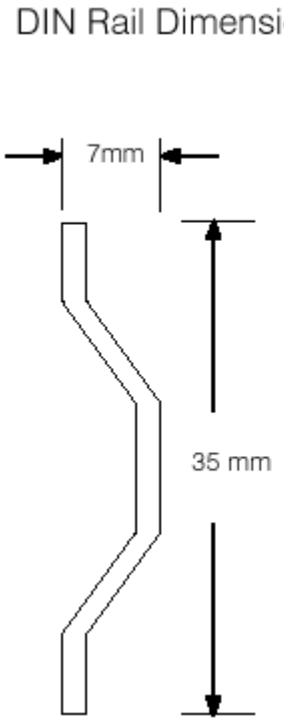
安装尺寸

DL05 单元尺寸如下所示，其安装可以用螺钉安装或使用 35mm 标准导轨安装。

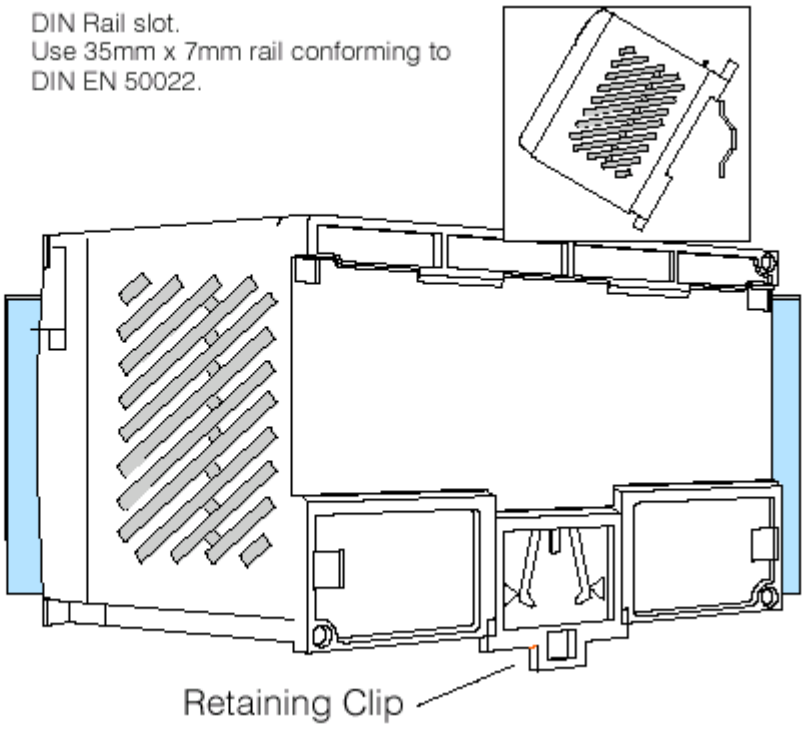


导轨安装

DIN 导轨尺寸

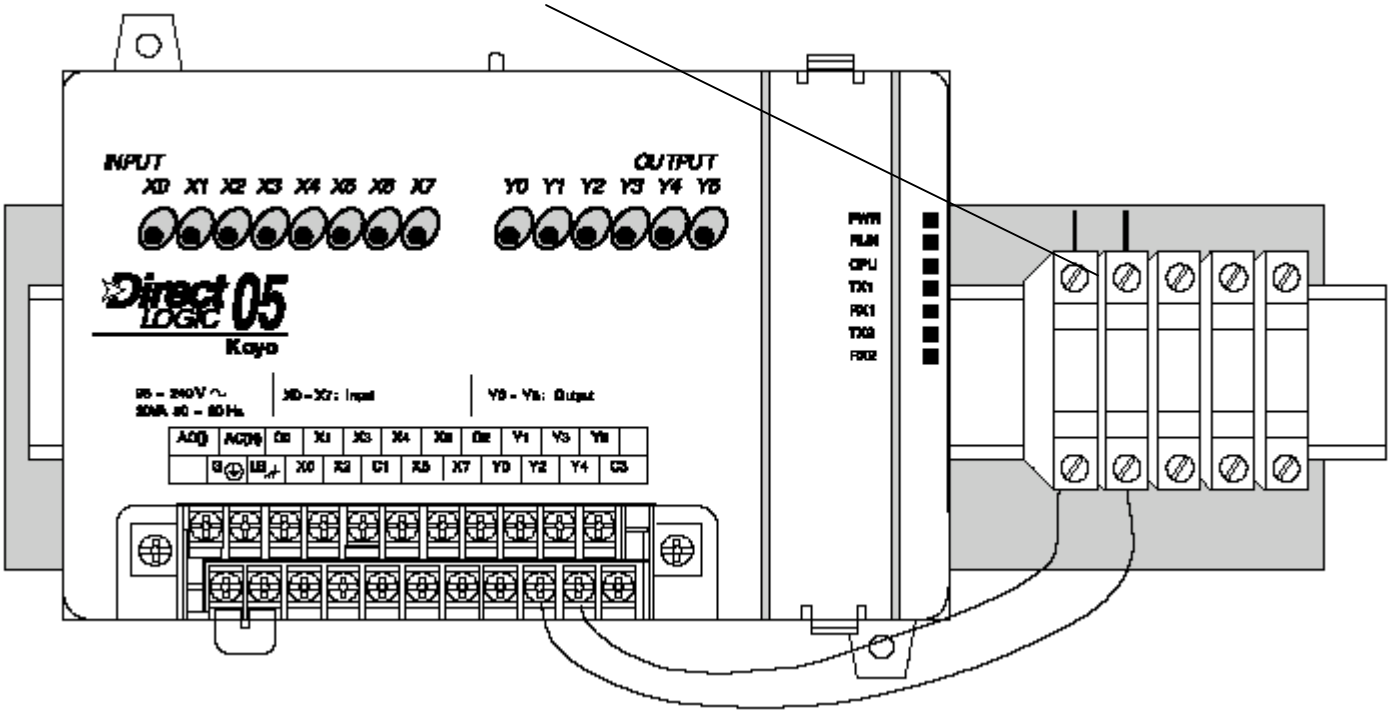


DIN 导轨槽
使用 35mm×7mm 导轨



输入/输出回路保险丝

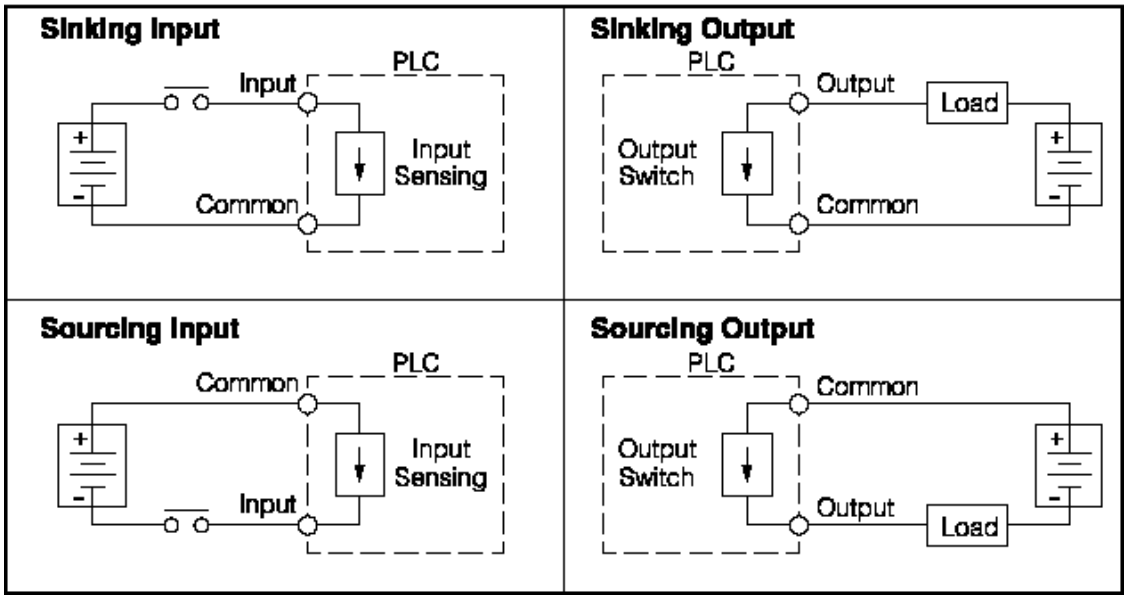
DL05 系列 PLC 的输入和输出回路没有内部保险丝。为了保护 PLC，我们建议外加保险丝。



源/汇的概念

Sink: 汇

Sourcing: 源



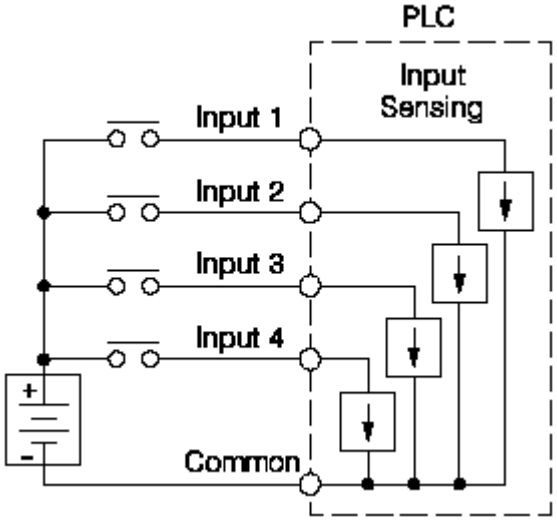
I/O 公共端"Common"的概念

I/O 公共端"Common"的概念

PLC 的 I/O 形成回路，必须有一个端子流入电流，另一个端子流出电流。因此每个 I/O 点至少需要 2 个端子相连。

如果有足够的空间和预算，完全可以为每个 I/O 点提供 2 个端子。但是，这种配置是不切实际或在大部分应用中是不必要的。因此在实际设计中，一般为多个输入或多个输出点合成一组，公用一个端子。例如右图中 4 个输入点公用 1 个端子，因此 4 路输入仅需要 5 个端子。

注意：上面图中，如果所有输入同时动作，公共端的电流将是每个回路的 4 倍。对输出回路这点尤其重要，一般在公共端需要更大负载能力的接线。



下图是完整的接线端子，2 组 4 输入点组合、2 组 3 输出点组合。每组 1 个公共端。各组间相互电气隔离。

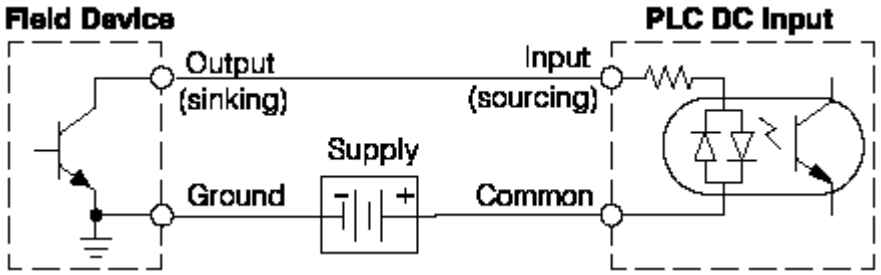
DC 输出型，所有输出点合用一个公共端，右下端的端子为输出点用外部电源端子。

连接 DC 型 I/O 与晶体管现场设备(“ Solid State ”)

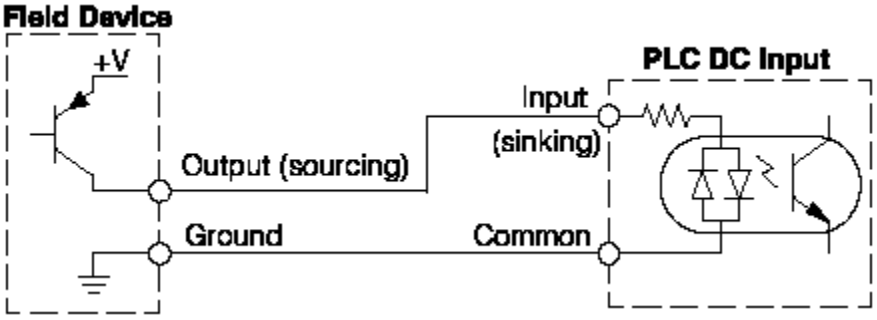
DC 型 I/O 回路有时只允许一种电流方向。对一些晶体管接口的现场设备一般也是这样的。但现场设备有可能是源或汇。当两个设备连接成 DC 回路时，一个必须是源而另一个必须是汇。

晶体管传感器输入

DL05 的 DC 形输入可以判别这两种方向的电流，因此既可以接成源也可以接成汇。下面的图中，现场设备的集电极开路 NPN 输出。它汇集从 PLC 输入点来的电流（即源电流）。需要+12VDC 或+24VDC 电源。



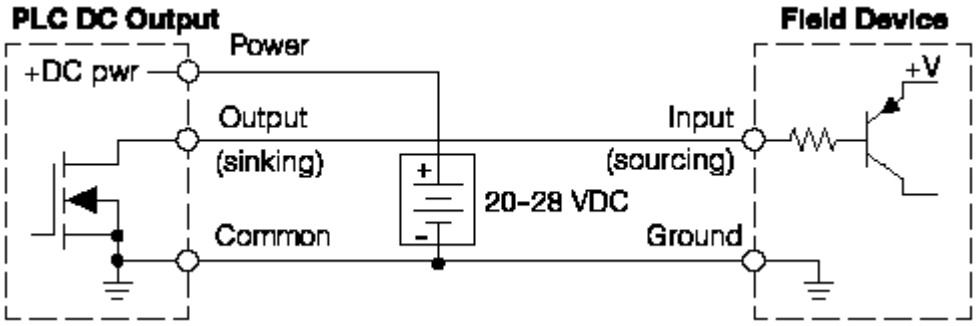
接下来的图中，现场设备为射极开路 PNP 输出。它提供到 PLC 输入点的电流，然后汇集回到地。因此现场设备是源电流，不需要附加电源。



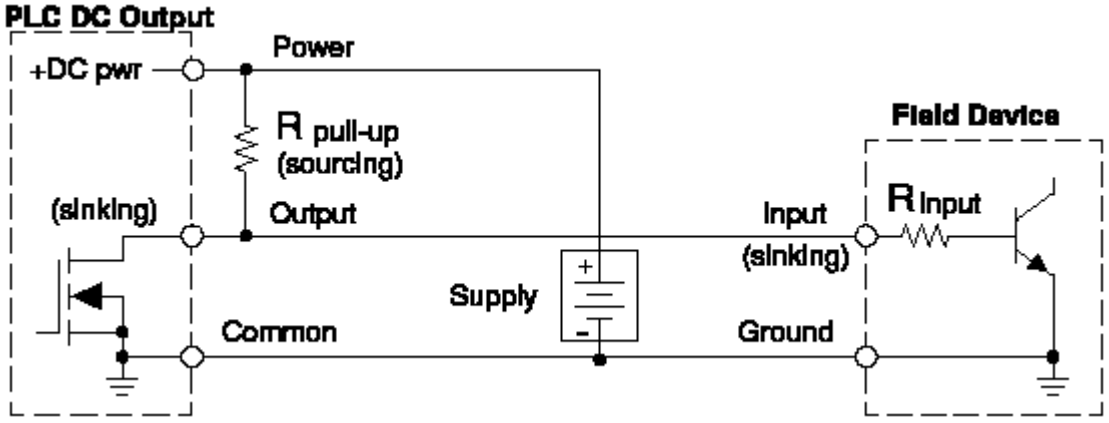
晶体管输出

有时需要连接 PLC 输出点到设备的晶体管输入。这种连接通常传送低电平信号而不是功率驱动。
DL05 的 DC 输出只有汇输出，这意味着每个输出点提供一条到地的通道；并且 6 个输出点是 1 个电气公共端（有 2 个螺丝接线端子）。另外 DC 输出需要外加 20-28VDC 工作电源。

下图中，PLC 输出点动作时汇集电流到公共端。它与现场设备的源形输入点连接。



接下来的图中，我们可以连接 PLC 的 DC 输出到现场设备的汇输入。这是一种小变通，因为 PLC 输出和现场设备输入都是汇型。而每个回路必须有 1 个源和 1 个汇设备，因此需要增加 1 个上拉电阻到 PLC 的输出部分使之具有源驱动能力。在下图中，在输出点和 DC 回路电源之间连接 $R_{pull-up}$（上拉电阻）。



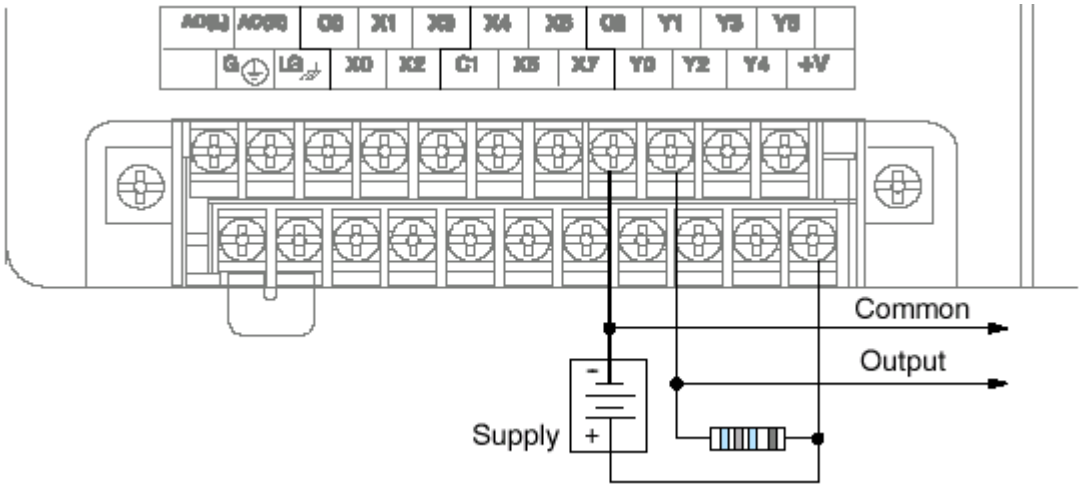
注意：
1、不要尝试用上拉方法驱动大的负载（>25mA）。
2、电阻反转了输出逻辑。即（从 PLC 程序侧看）当 PLC 输出为 OFF 时现场设备有输入。

选择 $R_{pull-up}$（上拉电阻）的大小是很重要的。计算时，需要知道当输入动作时现场设备的通常输入电流 I_{input}。如果不知道可以计算出来（典型值 15mA）。然后用 I_{input} 和外部电源电压计算出 $R_{pull-up}$、计算出功率 $P_{pull-up}$（瓦）。

$$I_{input} = \frac{V_{input\ (turn-on)}}{R_{input}}$$

$$R_{pull-up} = \frac{V_{supply} - 0.7}{I_{input}} - R_{input} \qquad P_{pull-up} = \frac{V_{supply}^2}{R_{pullup}}$$

下图是 DL05 PLC 应用上拉电阻的实际接线方法。



继电器输出接线方法

继电器输出接线方法

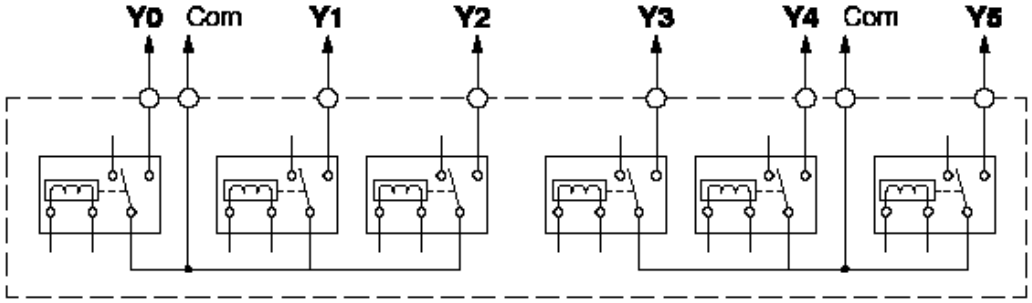
D0-05AR 和 D0-05DR 型号是继电器输出。比较适合以下场合应用：

- 负载需要大的电流超过晶体管输出驱动能力
- 变化不快的场合
- 某些输出需要和其他输出分离开来，例如一些负载需要 AC 而另一些需要 DC。

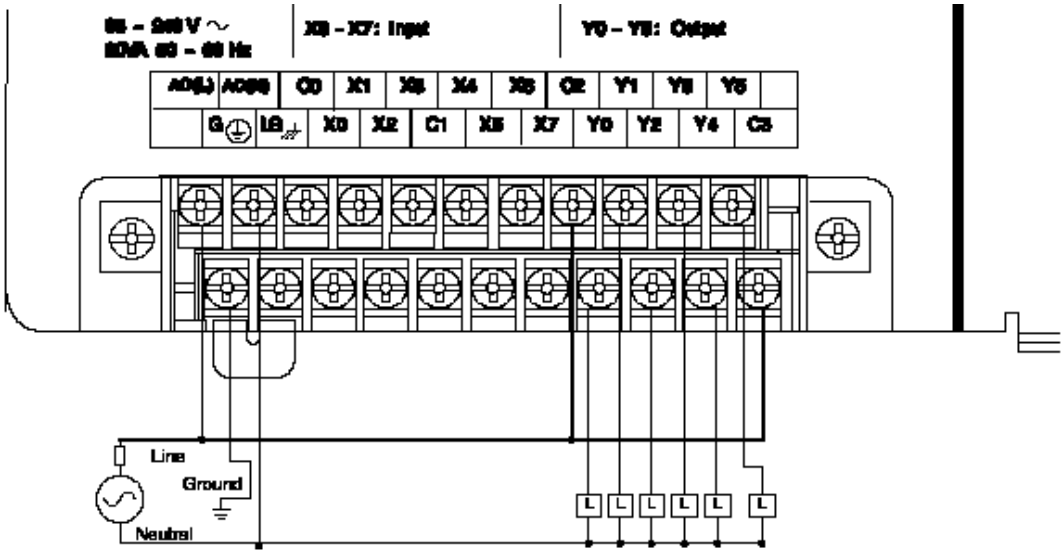
但不适合以下场合：

- 负载电流小于 10mA
- 负载需要快速开关

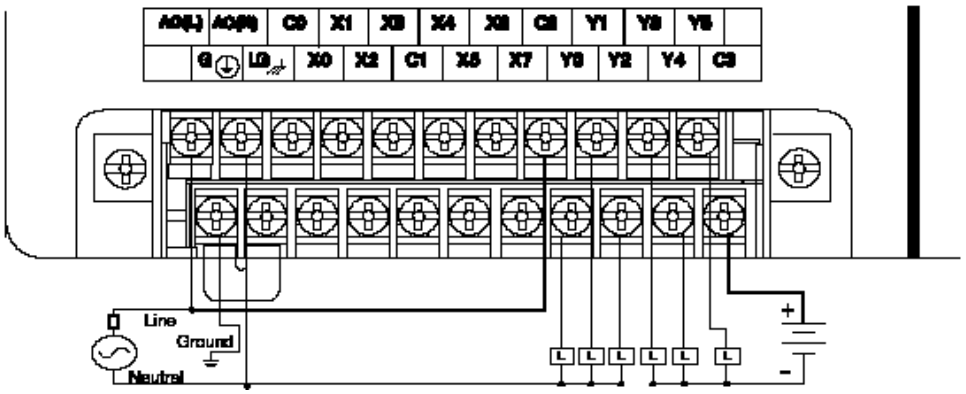
假如继电器输出是适合需要的，这里介绍继电器输出与负载的接线方法。DL05 有 6 个带常开接点的继电器输出，每 3 个为 1 个公共端。下图是 PLC 内部的继电器接线。注意，每组输出是电气分离的。



下图是所有负载与 DL05 PLC 使用同一 AC 电源的例子，其中所有公共端是连接在一起的。



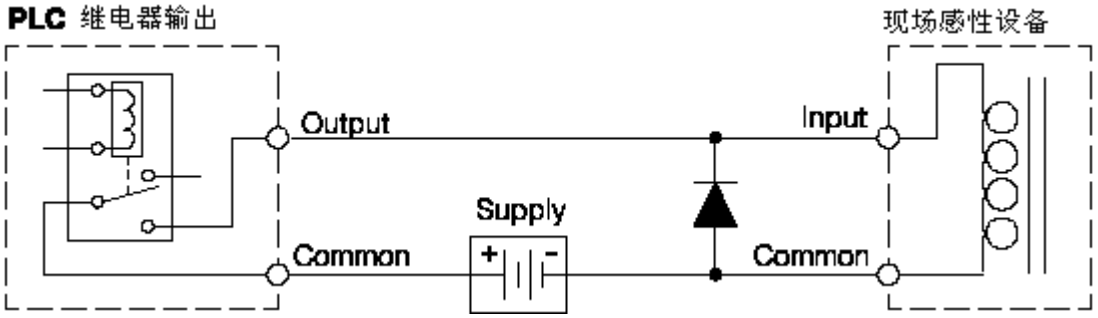
而在下图中，Y0-Y2 与 DL05 PLC 使用相同的 AC 电源，Y3-Y5 使用独立的 DC 电源。这样，公共端是独立的，可以用于独立负载方式。



延长继电器触点寿命

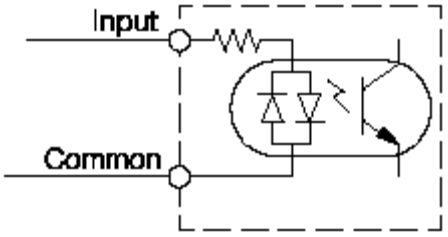
继电器触点的寿命与继电器开关的次数、频率、开关时产生的瞬间放电大小等因素有关。你可以采用多种方法来帮助延长继电器的寿命，例如：仅在需要时开关继电器；尽可能在小电流状态下开关负载；在驱动感性负载时，想法抑制其产生的诱导电流等等。

在 DC 回路中驱动感性负载时，可在靠近感性负载的位置增加二极管来保护继电器（在 AC 回路中禁用此法！）。但注意不要用一般小信号的二极管，并且方向不要搞错，否则当继电器动作时变成了短路。

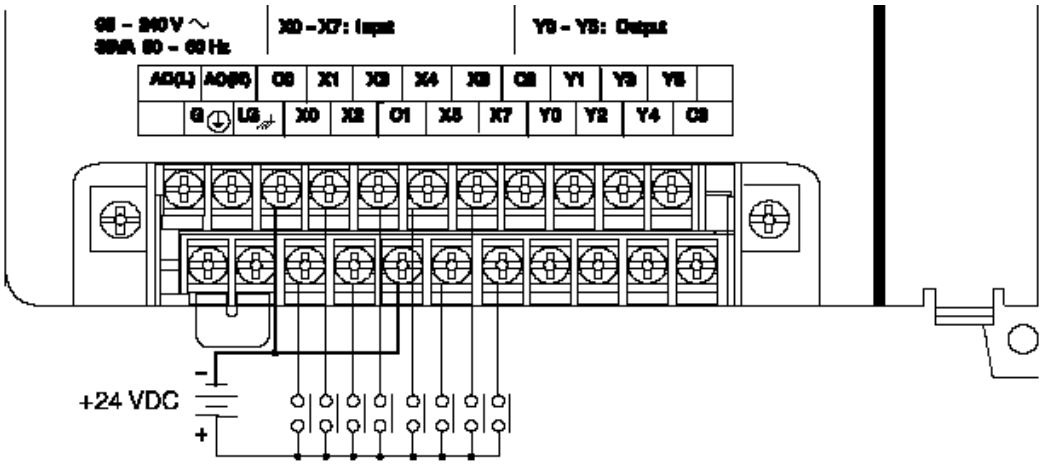


DC 输入接线方法

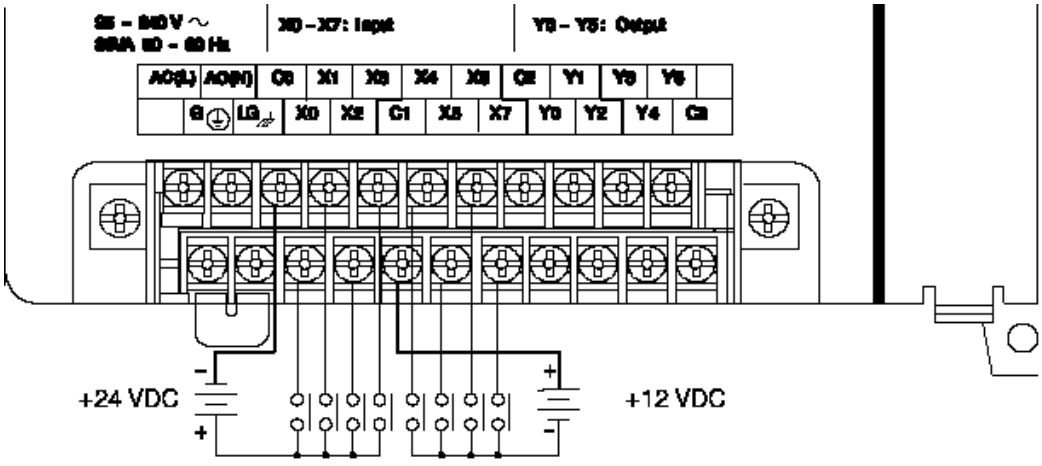
DL05 系列 PLC 的 DC 输入可以是源或汇方式。因此可以将一半输入接成源而另外一半接成汇。但接在一个公共端上的输入必须相同。



在下面最简单的例子图中，所有公共端连接在一起、所有输入连接成汇点方式。



在下图中，前 4 个输入点是汇点输入，后 4 个输入点是源输入。

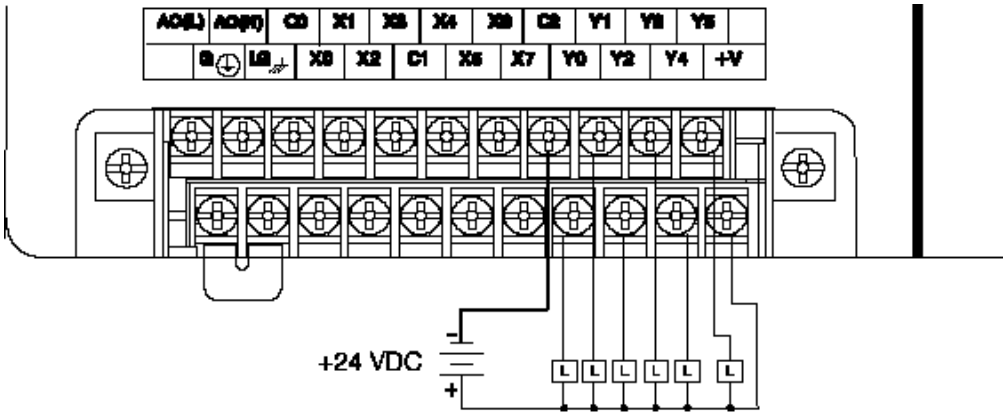


DC 输出接线方法

DL05 DC 输出部分采用高性能、低阻抗高速晶体开关管。它具有以下输出特点：

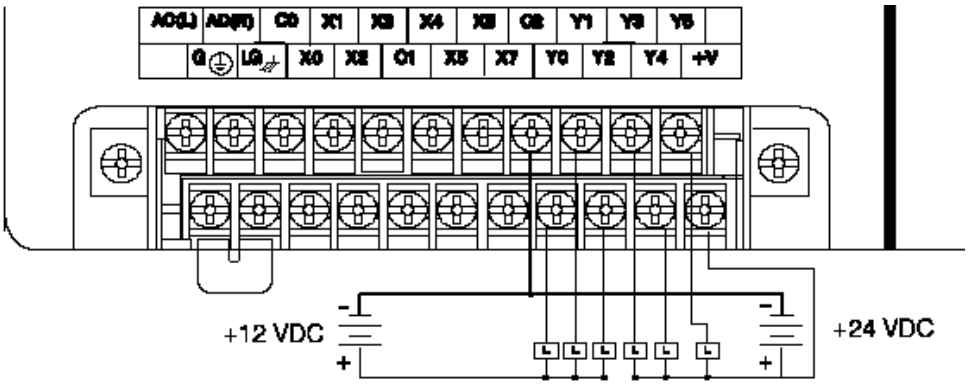
- 所有 6 个输出点只有 1 个电气公共端，为 1 组。
- 输出是电流汇点方式。但不同的输出端子可以使用不同的 DC 电源电压。
- PLC 内部输出回路需要外部电源。电源(-)端必须连接到公共端，而电源(+)端连接到最右边的端子。

下图中，所有 6 个输出点为一个公共端。



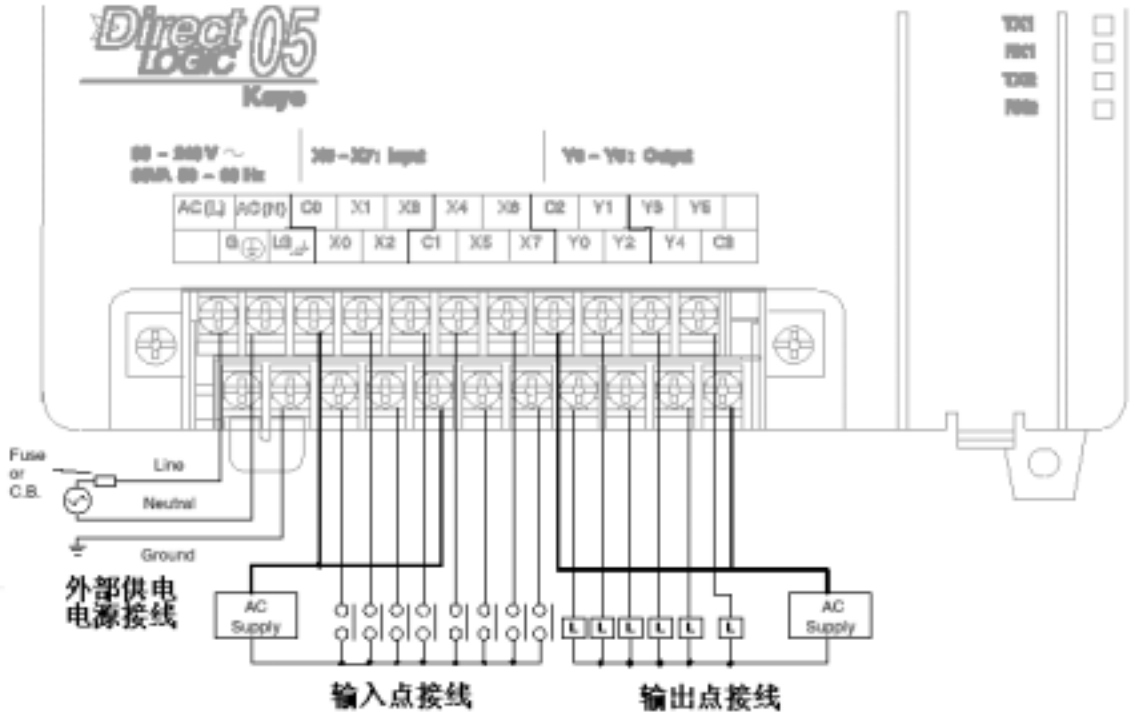
在下图中，输出使用了不同的电源。前 3 个输出使用+12VDC，后 3 个输出使用+24VDC。当然，也可以用任意多种电源，只要符合：

- 所有电压都在允许范围内
- 所有输出点接线成汇点方式
- 所有电源(-)端连接在一起



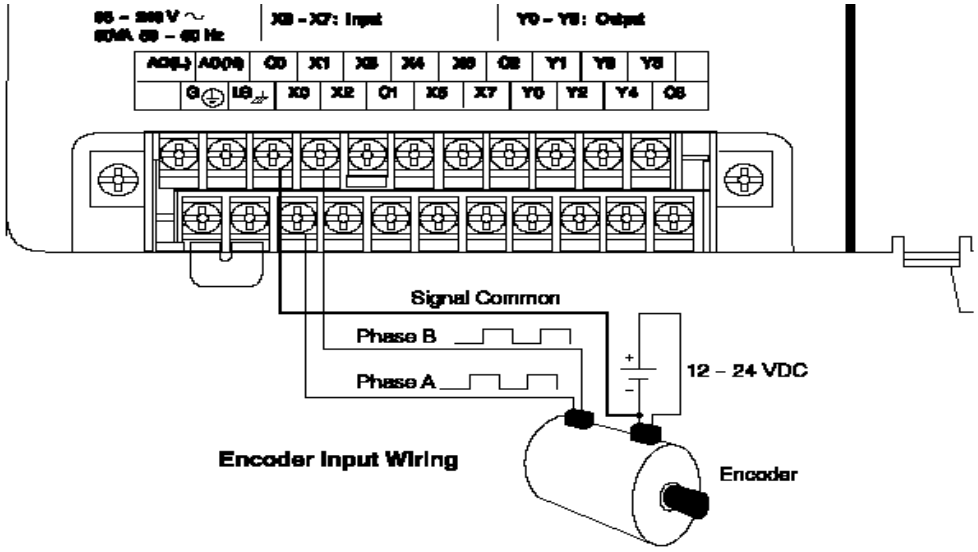
AC 输入/AC 输出接线方法

DL05 有交流型的输入/输出，下图为 D0-05AA 的接线示意图；图中外部供电电源采用其左边的 4 个接线端子。**注意，AC 型外部供电电源可用 AC110V 或 AC220V；输入只能为 AC110V；输出可为 17-240VAC。**

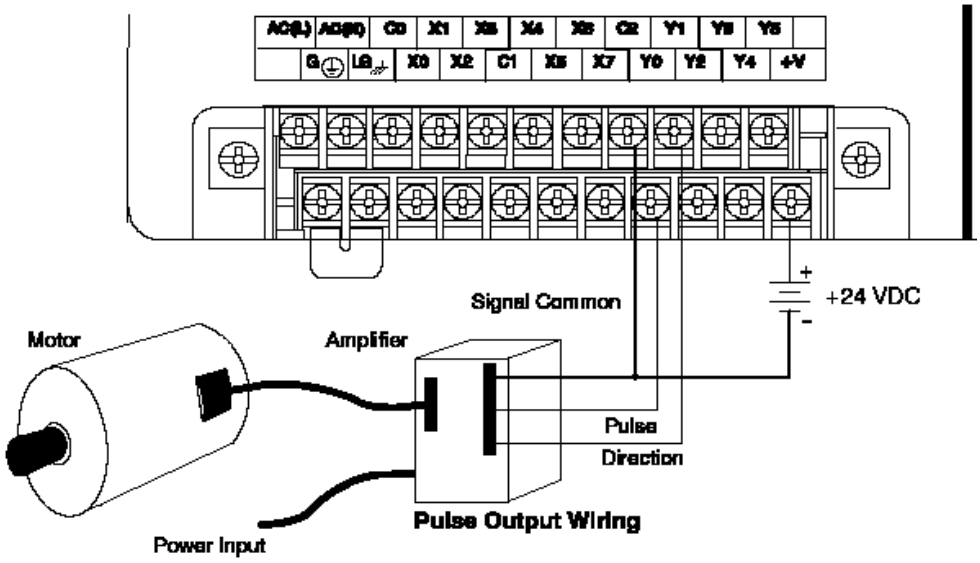


高速 I/O 接线方法

DL05 的 DC 输入或输出可以使用高速 IO（HSIO）功能。回路配置是可编程改变的，并且选择的 I/O 点处理是独立于 CPU 扫描的。第 4 章介绍了 HSIO 的 6 种模式，这里介绍最常用的 2 种接线。高速输入接口 X0-X2，正确配置后，可以按下图连接 A、B 相编码器进行加/减计数。



DC 输出点可以用作高速脉冲输出，产生高速脉冲信号控制步进电机/智能驱动系统。输出点 Y0 和 Y1 可以配置为产生脉冲+方向信号，或者 CCW+CW 脉冲。



DL05 I/O 电气性能规格

DC 输入电气性能（I0~I7）

项目	高速输入点 I0~I2	普通输入点 I3~I7
输入电压范围	10.8~26.4 VDC	10.8~26.4 VDC
工作电压范围	12~24 VDC	12~24 VDC
峰值电压	30 VDC(最高频率 5kHz)	30 VDC
最小脉冲宽度	100 μs	N/A
ON 电平	>10 VDC	>10 VDC
OFF 电平	<2.0 VDC	<2.0 VDC
输入阻抗	1.8KΩ 在 12 ~24 VDC	2.8KΩ 在 12 ~24 VDC
最大输入电流	6mA 在 12 VDC, 12mA 在 24 VDC	4mA 在 12 VDC, 8.5mA 在 24 VDC
最小 ON 电流	>5mA	>4mA
最大 OFF 电流	<0.5mA	<0.5mA
OFF 到 ON 延时	<100 μs	2~8ms,典型 4ms
ON 到 OFF 延时	<100 μs	2~8ms,典型 4ms
状态指示灯	逻辑端	逻辑端
公共端子	4 通道/公共端 2 组	

AC 输入电气性能（I0~I7）

项目	规格
输入电压范围	80~132 VAC , 47~63Hz
工作电压范围	90~120 VAC, 47~63Hz
输入电流	8mA 在 100 VAC 50Hz 10mA 在 100 VAC 60Hz
最大输入电流	12mA 在 132 VAC 50Hz 15mA 在 132 VAC 60Hz
输入阻抗	14KΩ 在 50 Hz, 12KΩ 在 60 Hz
ON 电流/电压	>6mA
OFF 电流/电压	<2mA 在 20VAC
OFF 到 ON 延时	<40ms
ON 到 OFF 延时	<40ms
状态指示灯	逻辑端
公共端子	4 通道/公共端 2 组

DC 输出电气性能（Q0~Q5）

项目	脉冲输出点 Q0，Q1	普通输出点 Q2~Q5
输出电压范围	5~30 VDC	5~30 VDC
工作电压范围	6~27 VDC	6~27 VDC
峰值电压	<50 VDC(最高频率 7kHz)	<50 VDC
导通电压降	0.3VDC 1A	0.3VDC 1A
最大电流（阻性）	0.5A/点（作普通输出点时 1A/点）	1A/点
最大漏电流	15μA 30VDC	15μA 30VDC
最大浪涌电流	2A 100mS	2A 100mS
外部工作电源	20~28VDC 最大 150mA	20~28VDC 最大 150mA
OFF 到 ON 延时	<10μs	<10μs
ON 到 OFF 延时	<30μs	<60μs
状态指示灯	逻辑端	逻辑端
公共端子	6 通道/公共端 1 组	
保险丝	无	

继电器输出电气性能（Q0~Q5）

项目	规格
输出电压范围	5~264 VAC (47~63Hz), 5~30VDC
工作电压范围	6~240 VAC(47~63Hz), 6~27VDC
输出电流	2A/点，6A/公共点
最大电压	264VAC,30VDC
最大漏电流	0.1mA 在 264 VAC
最小推荐负载	5mA @ 5VDC
OFF 到 ON 延时	<15ms
ON 到 OFF 延时	<10ms
状态指示灯	逻辑端
公共端子	3 通道/公共端 2 组
保险丝	无（推荐外加）

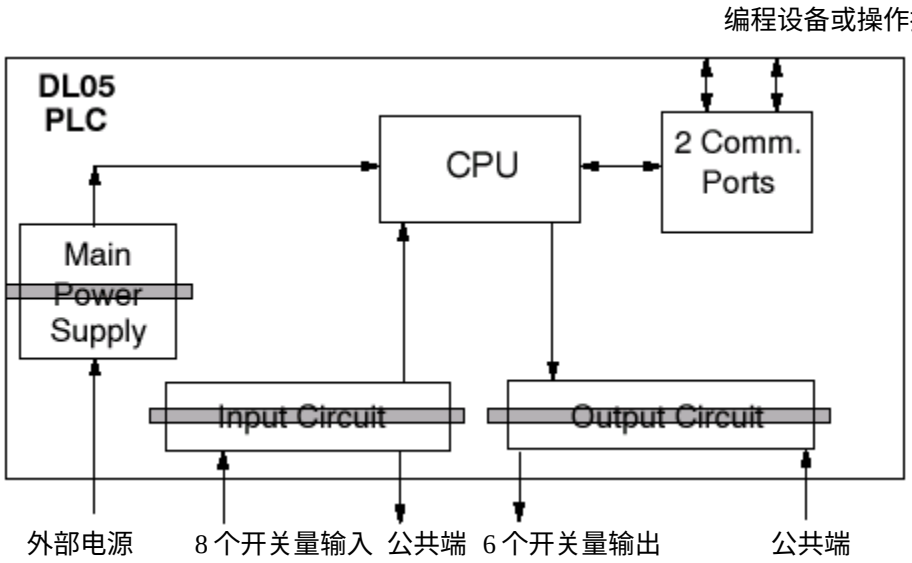
AC 输出电气性能(Q0~Q5)

项目	规格
输出电压范围	15~264 VAC (47~63Hz)
工作电压范围	17~240 VAC (47~63Hz)
ON 电压降	1.5VAC(>50mA) 4.0 VAC(<50mA)
最大电流	0.5A/点，1.5A/公共点
最大漏电流	<4mA @ 264VAC
最大涌入电流	10 A for 10ms
最小负载	10mA
OFF 到 ON 延时	1ms
ON 到 OFF 延时	1ms＋1/2 周期
状态指示灯	逻辑端
公共端子	3 通道/公共端 × 2 组
保险丝	无（推荐外加）

第 3 章 CPU 性能和操作

简介

CPU 是 PLC 的心脏，几乎所有的操作都由 CPU 控制。DL05PLC 的组成如下所示。



注意：
使用 DL05 的高速 I/O 功能（HSIO）需要特别的设置，具体请参见第四章。

DL05 特点

DL05 为固定式 PLC，具有 8 点输入点，6 点输出点，其 I/O 点数不可扩展；

DL05 具有 2K 语程序空间和 4K 字数据存储器空间，程序采用 FlashROM 存储器；数据寄存器内容、系统参数区数据和其他信息由大容量电容保持，即使在掉电情况下，其内容也可保持几个小时，需要长久保存其中数据，可选择扩充单元选购件；

DL05 提供多达 120 多条指令，适合各种应用场合的编程控制；

DL05 内建 2 个 RS232C 通讯口，可同时连接编程设备和各种通讯设备，方便调试、使用。

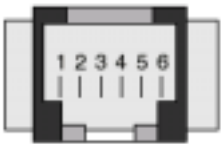
CPU 规格

项目	DL05
控制方式	循环扫描、中断
编程方式	指令语，梯形图、级式共用
梯形图程序容量（语）	2048
全部数据寄存器 R（字）	4096
用户使用数据寄存器（字）	3968
不挥发数据寄存器（字）	128
特殊寄存器 R	128
布尔指令执行	2.0μS
典型扫描（布尔）	2.7-3.2mS
运行中修改程序	Yes
扫描	可变/固定
手持编程器	Yes
Windows 环境编程软件	Yes
内装 RS232C 通讯口	2 个
FLASH 存储器	CPU 标准
本体开关量	8 输入，6 输出
本体模拟量	None
高速 I/O	Yes，2
指令数	129
输入 继电器 I	256
输出继电器 Q	256
中间继电器 M	512
特殊继电器 SP	512
级 S	256
定时器 T	128
计数器 C	128
立即 I/O	Yes
中断输入（外部/定时）	Yes
For/Next 循环	Yes
数学运算	整数
鼓型控制	Yes
日历/时钟	No
内部诊断	Yes
口令保护	Yes
系统错误履历	No
用户错误履历	No
后备电池	No(超级电容) Yes，用 Mem 卡

CPU 硬件设置

通讯口引脚图

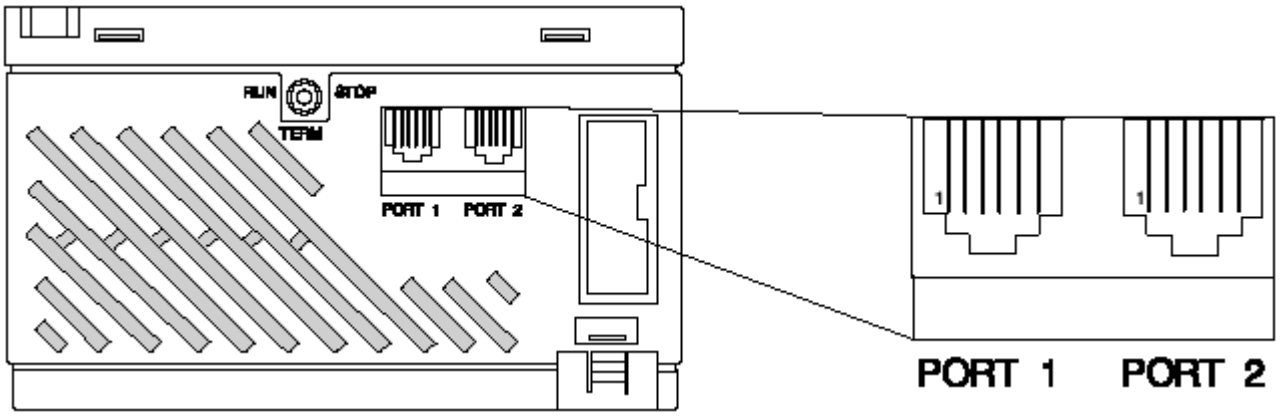
DL05 PLC 有 2 个 RS232C 通讯口。Port1 可用于(连接)手持编程器、编程软件、操作员接口，MODBUS 子局、CCM 子局，固定 BPS 为 9600。Port2 可用于(连接)手持编程器、编程软件、操作员接口，MODBUS 主/子局、CCM 主/子局，BPS 可设定 300~38.4k。



6 芯电话插座

Port 1 Pin Descriptions		
1	0V	Power (-) connection (GND)
2	5V	Power (+) connection
3	RXD	Receive Data (RS232C)
4	TXD	Transmit Data (RS232C
5	5V	Power (+) conection
6	0V	Power (-) connection (GND)

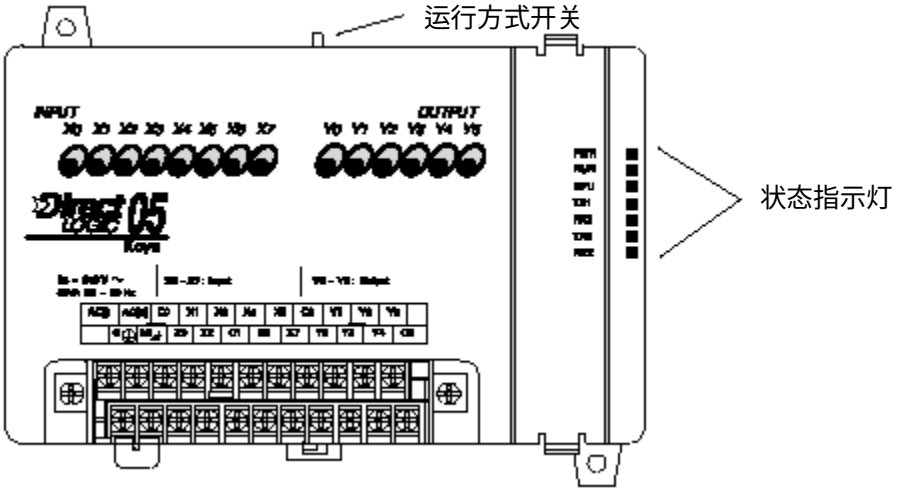
Port 2 Pin Descriptions		
1	0V	Power (-) connection (GND)
2	5V	Power (+) connection
3	RXD	Receive Data (RS232C)
4	TXD	Transmit Data (RS232C
5	RTS	Request to Send
6	0V	Power (-) connection (GND)



在进行实际的编程工作之前，有一些基本的设置需要设置，最基本的如下：

- 选择或改变 PLC 运行方式；
- 使用编程设备（手持编程器、计算机编程软件）菜单功能；
- 清除程序（和其他存储区域）；
- 初始化系统参数存储器；
- 设置停电保持数据范围。

下面对这些基本的设置内容进行说明，下图为 DL05PLC 前面板图。



状态指示灯

在 PLC 右侧的状态指示灯有特殊的意义，在编程和调试时可以参考。

指示灯	状态	意思
PWR	ON	电源正常
	OFF	电源异常
RUN	ON	ＰＬＣ在运行中
	OFF	ＰＬＣ在停止或编程状态
CPU	ON	ＣＰＵ自检出错
	OFF	ＣＰＵ自检正常
TX1	ON	在通讯口１上有数据传出
	OFF	在通讯口１上无数据传出
RX1	ON	在通讯口１上有数据接受
	OFF	在通讯口１上无数据接受
TX2	ON	在通讯口２上有数据传出
	OFF	在通讯口２上无数据传出
RX2	ON	在通讯口２上有数据接受
	OFF	在通讯口２上无数据接受

另面板上的 I/O 状态指示灯用于监视某一时刻的 I/O 通断状态。

运行方式开关

ＤＬ０５系列ＰＬＣ具有一个运行方式开关，只有当其处于ＴＥＲＭ位置时，才允许通过编程设备对其进行操作。该开关在各个位置时允许的操作如下表。

运行开关位置	PLC 功能
RUN	PLC 进入强制运行状态，不允许编程设备操作
TERM	允许编程设备操作方式，此时允 PLC 进入运行、停止和试运行方式
STOP	PLC 进入强制停止状态，不允许编程设备操作

有 2 种方式可改变 PLC 运行状态，一是使用运行开关控制；二是在运行开关处于 TERM 位置时，通过编程设备改变 PLC 运行方式。编程设备有手持式编程器和计算机用编程软件 2 种，有关编程设备的操作，请参见相关编程器操作手册、或编程软件随机资料。

当运行开关处于 TERM 位置时，每当 DL05PLC 上电运行时，它是根据其最近停电时记忆的运行状态而定，但要注意，如果放置时间过长，则再次上电时，PLC 的运行状态不定。

当运行开关处于 RUN 或 STOP 位置时，则 PLC 强制进入 RUN 或 STOP 状态。

系统参数区

在 DL05PLC 上有一存储区域用于设置 PLC 系统的基本参数，其主要包括：程序名、口令、暂停参数、停电保持参数、通用口通讯参数、监控定时器等等，系统参数区主要参数如下表：

项目		初始值	可设定范围
程序名		未登记	8 位以内英、数字
口令		00000000（未登记）	8 位数字（BCD）
暂停参数		全 OFF	ON/OFF（Q 领域）
停电保持区域	内部继电器 M	M400~M777	M0~M777
	数据寄存器 R	R1400~R7777	R0~R7777
	定时器 T	无	T0~T177
	计数器 C	C0~C177	C0~C177
	级 S	无	S0~S377
监控定时器 WDOGT		200ms	2~9998ms

停电保持区域

DL05PLC 出厂配置中是不带电池的，其停电保持区域内容中的内容是通过大容量电容来保持的，不能长久停电保持，如要长久保持，请选购存储器单元 D0-01MC。

DL05 功能存储器

存储器类型	开关量引用(八进制)	寄存器引用(八进制)	数量 (十进制)
输入①	0-377	R40400-40417	256
输出①	0-377	R40500-40517	256
控制继电器	0-777	R40600-40637	512
特殊继电器	0-777	R41200-41237	512
定时器	0-177	R41100-41107	128
定时器经过值	无	R0-177	128
定时器状态位	0-177	R41100-41107	128
计数器	0-177	R41140-41147	128
计数器经过值	无	R1000-R1177	128
计数器状态位	0-177	R41140-41147	128
数据寄存器	无	R1200-7377	3968
非易失寄存器	无	R7400-7577	128
级	0-377	R41000-41017	256
系统参数	无	R7600-7777	128

①、DL05 系统硬件预装 8 个输入和 6 个输出，但地址存在。

美国 ADC 公司和光洋电子（无锡）有限公司的功能存储器代码对照表

项目	美国 ADC	光洋无锡
输入	X	I
输出	Y	Q
中间继电器	C	M
特殊继电器	SP	SP
定时器	T	T
计数器	CT	C
级	S	S
数据寄存器	V/P	R/P

DL05 系统参数寄存器

对于下表中的数据寄存器，在 DL05 中，已预先规定了其用途，请不要再挪作它用。

系统参数寄存器	说明	默认值/范围
R2320-2377	默认高速计数多段预置值	N/A
R7620-7627	S-10D 操作单元占用	
R7620	数据设定寄存器号	R0-2377
R7621	注释显示寄存器号	R0-2377
R7622	寄存器数	1-16
R7623	数据显示寄存器号	R0-2377
R7624	文字显示寄存器号	R0-2377
R7625	键分配	
R7626	上电方式	0，1，2，12，3
R7627	口令	0000-9999
R7630	通道 1 多段设定寄存器号。默认是 2320，因为是 24 个预置可能，默认范围是 R2320-2377。	默认：R2320 范围：R0-R2320
R7631-7632	保留	
R7633	设置高速计数、中断、脉冲捕捉、脉冲输出和输入滤波功能。(高字节用于上电时选择 RUN 方式)	默认：0060 低字节范围： 10-计数 20-加减计数 30-脉冲输出 40-中断 50-脉冲捕捉 60-滤波输入 高字节： Bits 8~12,14,15 不使用； Bit 13 设置上电时 RUN 方式(仅当方式开关在 TERM 位置时)
R7634	I0 高速计数功能设定	默认：1006
R7635	I1 高速计数功能设定	默认：1006
R7636	I2 高速计数功能设定	默认：1006
R7637-7646	保留	
R7647	定时中断	默认：0000 范围：0003-03E7H (3-9999ms)
R7650-7654	保留	
R7655	Port2：协议、通讯超时和应答延时	默认：00E0
R7656	Port2：局号、波特率、停止位和校验	默认：8501
R7657	Port2：设置结束码，参数设置完了的通知	默认：0A00

系统参数寄存器	说明	默认值/范围
R7660	扫描控制：保持扫描控制模式	默认：0000
R7661	设置定时超过的计数：对扫描超过用户设置时间的次数计数	N/A
R7662-7717	保留	N/A
R7720-7722	DV-1000 操作接口占用	N/A
R7720	定时器设定开始寄存器号	N/A
R7721	计数器设定开始寄存器号	N/A
R7722	高字节--定时器，低字节--计数器设定点数	N/A
R7723-7750	保留	N/A
R7751	FALT 指令执行的代码	N/A
R7752-7754	保留	N/A
R7755	错误代码—致命错误代码	N/A
R7756	错误代码—重大错误代码	N/A
R7757	错误代码—轻度错误代码	N/A
R7760-7762	保留	N/A
R7763	程序语法错误地址	N/A
R7764	语法错误代码	N/A
R7765	扫描计数	N/A
R7766-7774	保留	N/A
R7775	扫描时间-当前(ms)	N/A
R7776	扫描时间-最小(ms)	N/A
R7777	扫描时间-最大(ms)	N/A

DL05 特殊继电器

“**特殊继电器**”是 **CPU** 操作系统设置的，用来指示所发生的特别系统事件的触点。在你的程序中可使用这些触点。了解用于特别状态下的特殊继电器触点的正确作用，可以节省编程时间。因为操作系统可设定和清除特殊继电器触点，所以在程序中，它们仅作输入接点用。

启动和实时继电器

SP0	初始复位	只在一个电源周期或程序-运行转换后，第一次扫描 **on**。在第二次扫描时，继电器被复位成 **off**。当某个功能需在程序启动时被执行时，可以使用它。
SP1	常时 **ON**	提供一个触点，保证每次扫描时某个指令均被执行。
SP3	**1** 分钟时钟脉冲	**30** 秒 **on**，**30** 秒 **off**。
SP4	**1** 秒钟时钟脉冲	**0.5** 秒 **on**，**0.5** 秒 **off**。
SP5	**100ms** 时钟脉冲	**50ms on**，**50ms off**。
SP6	**50ms** 时钟脉冲	**25ms on**，**25ms off**。
SP7	扫描时钟脉冲	由 **on** 状态开始。

CPU 状态继电器

SP11	强制 **run** 状态。	当 **mode** 开关在 **run** 位置，**CPU** 正在运行时 **on**。
SP12	**Terminal run** 状态。	当 **CPU** 在 **run** 状态时 **on**。
SP13	**Test run** 状态。	当 **CPU** 在 **test run** 状态时 **on**。
SP15	**Test stop** 状态。	当 **CPU** 在 **test stop** 状态时 **on**。
SP16	**Terminal PGM** 状态。	当 **mode** 开关在 **TERM** 位置，**CPU** 在 **program** 状态时 **on**。
SP17	强制 **stop**。	当 **mode** 开关在 **STOP** 位置时 **on**。
SP20	强制 **stop** 状态。	**STOP** 指令执行后 **on**。
SP22	中断许可	由 **ENI** 指令允许中断后 **on**。

系统监控

SP36	替代设置继电器	使用替代功能时 **on**。
SP37	扫描控制错误	实际扫描时间超过规定的扫描时间时 **on**。
SP40	重度异常	发生如 **I/O** 通讯失败等重度异常时 **on**。
SP41	轻度异常	发生轻度错误时 **on**。
SP42	诊断错误	发生诊断错误或系统错误时 **on**。
SP44	**Program memory** 异常	存储器发生异常时 **on**。
SP45	**I/O** 异常	发生诸如保险丝熔断等 **I/O** 错误时 **on**。
SP46	通信异常	在 **CPU** 的任一口上发生通讯异常时 **on**。
SP50	错误指令	执行错误指令后 **on**。
SP51	监控定时器超时。	若 **CPU** 的监控定时器超时时 **on** 。
SP52	语法错误	当 **CPU** 处于运行中或进行语法检查时发现有语法错误，则 **on**。**V7755** 将保留错误码。
SP53	运算错误	**CPU** 无法进行运算处理时 **on**。
SP54	通讯错误	执行 **RX**，**WX**，**RD**，**WT** 指令发生错误时 **on**。
SP56	表指令溢出	当执行表指令时，若表的指针值超过表的范围，则 **on**。

累加器状态

SP60	小于标志继电器	累加器值小于指令值时 **on**。
SP61	等于标志断电器	累加器值等于指令值时 **on**。
SP62	大于标志继电器	累加器值大于指令值时 **on**。
SP63	零标志继电器	指令运算结果为零（在累加器中）时 **on**。
SP64	半借位标志继电器	当 **16** 位减法指令发生借位时 **on**。
SP65	借位标志继电器	当 **32** 位减法指令发生借位时 **on**。
SP66	半进位标志继电器	当 **16** 位加法指令发生进位时 **on**。
SP67	进位标志继电器	当 **32** 位加法指令发生进位时 **on**。
SP70	符号标志继电器	累加器中的值为负时 **on**。
SP71	指针指定出错标志	当指定了不存在的间接寄存器的区域时 **on**。
SP73	溢出标志继电器	带符号加、减法运算产生了错误的符号位，累加器有溢出时 **on**。
SP75	数据出错标志继电器	**BCD** 运算时，运算的数据不是 **BCD** 时 **on** 。
SP76	读零标志继电器	读入指令读入累加器的值为零时 **on**。

HSIO 脉冲输出继电器

SP104	控制结束	当脉冲控制被完成后 **on**。

通讯监控继电器

SP116	**CPU** 口 **2** 传递状态	当口 **2** 作数据传送时 **on**。
SP117	口 **2** 通讯错误	当口 **2** 通讯错误时 **on**。

HSIO Mode 10 计数器设定值一致继电器

SP540	当前值=目标值	计数器当前值等于 R**2320/R2321** 中的值时 **on**。
SP541	当前值=目标值	计数器当前值等于 **R2322/R2323** 中的值时 **on**。
SP542	当前值=目标值	计数器当前值等于 **R2324/R2325** 中的值时 **on**。
SP543	当前值=目标值	计数器当前值等于 **R2326/R2327** 中的值时 **on**。
SP544	当前值=目标值	计数器当前值等于 **R2330/R2331** 中的值时 **on**。
SP545	当前值=目标值	计数器当前值等于 **R2332/R2333** 中的值时 **on**。
SP546	当前值=目标值	计数器当前值等于 **R2334/R2335** 中的值时 **on**。
SP547	当前值=目标值	计数器当前值等于 **R2336/R2337** 中的值时 **on**。
SP550	当前值=目标值	计数器当前值等于 **R2340/R2341** 中的值时 **on**。
SP551	当前值=目标值	计数器当前值等于 **R2342/R2343** 中的值时 **on**。
SP552	当前值=目标值	计数器当前值等于 **R2344/R2345** 中的值时 **on**。
SP553	当前值=目标值	计数器当前值等于 **R2346/R2347** 中的值时 **on**。
SP554	当前值=目标值	计数器当前值等于 **R2350/R2351** 中的值时 **on**。
SP555	当前值=目标值	计数器当前值等于 **R2352/R2353** 中的值时 **on**。
SP556	当前值=目标值	计数器当前值等于 **R2354/R2355** 中的值时 **on**。
SP557	当前值=目标值	计数器当前值等于 **R2356/R2357** 中的值时 **on**。
SP560	当前值=目标值	计数器当前值等于 **R2360/R2361** 中的值时 **on**。
SP561	当前值=目标值	计数器当前值等于 **R2362/R2363** 中的值时 **on**。
SP562	当前值=目标值	计数器当前值等于 **R2364/R2365** 中的值时 **on**。
SP563	当前值=目标值	计数器当前值等于 **R2366/R2367** 中的值时 **on**。
SP564	当前值=目标值	计数器当前值等于 **R2370/R2371** 中的值时 **on**。
SP565	当前值=目标值	计数器当前值等于 **R2372/R2373** 中的值时 **on**。
SP566	当前值=目标值	计数器当前值等于 **R2374/R2375** 中的值时 **on**。
SP567	当前值=目标值	计数器当前值等于 **R2376/R2377** 中的值时 **on**。

DL05 网络系统设置

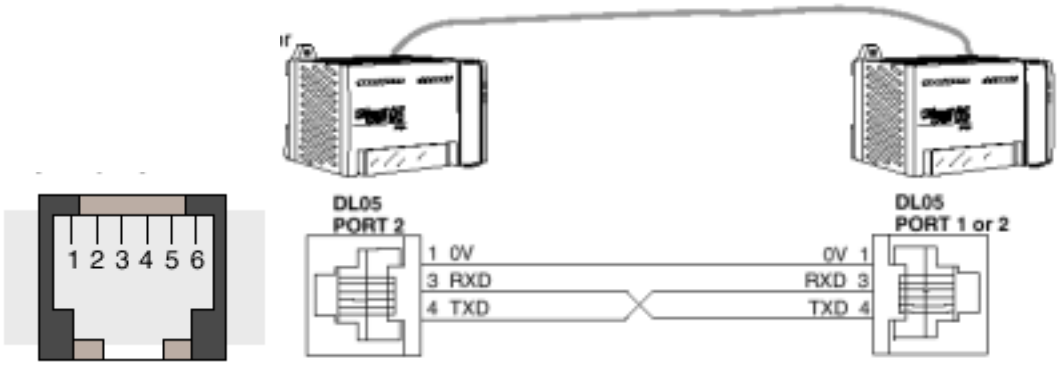
DL05 上的通用通讯口支持 MODBUS，CCM 通讯，可直接以 RTU 协议加入 MODBUS 网络；或加入 CCM 协议。当接入 MODBUS 网时，MODBUS 主局必须发出相应的读写指令。

通讯口 1	通讯口 2
可连接编程设备，操作面板等 6 芯电话接口，RS232C 通讯方式 通讯速度：9600BPS（固定） 奇偶校验：奇校验（固定） 局号：1 号局（固定） 8 位数据位，1 位起始位，1 位停止位 异步，半双工方式，DTE 协议：（自动选择） K 协议（仅子局功能）， CCM 协议（仅子局功能）， MODBUS 协议（仅子局功能）。	可连接编程设备，操作面板等 6 芯电话接口，RS232C 通讯方式 通讯速度：（ BPS） 300，600，1200，2400，4800，9600， 19200，38400 奇偶校验：奇校验（缺省），偶校验，无校验 局号：1 号局（缺省） 8 位数据位，1 位起始位，1 位停止位 异步，半双工方式，DTE 协议：（自动选择） K 协议（仅子局功能）， CCM 协议（主局/子局功能）， MODBUS 协议（主局/子局功能）， 无协议。

2 个通讯口的引脚分配如下表：

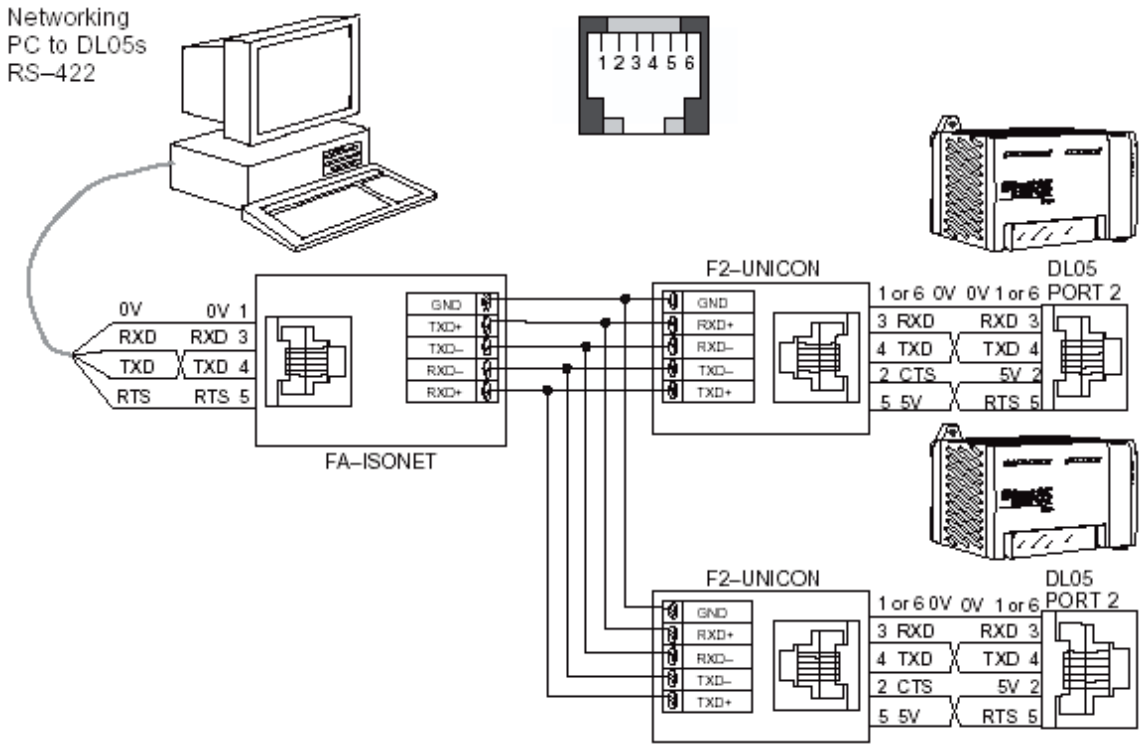
通讯口 1	通讯口 2
1 0V 电源地（GND）	1 0V 电源地（GND）
2 5V 电源正	2 5V 电源正
3 RXD 数据接受端	3 RXD 数据接受端
4 TXD 数据发送端	4 TXD 数据发送端
5 5V 电源正	5 RTS 发送请求端
6 0V 电源地（GND）	6 0V 电源地（GND）

Dl05 一对一连接（RS232C）

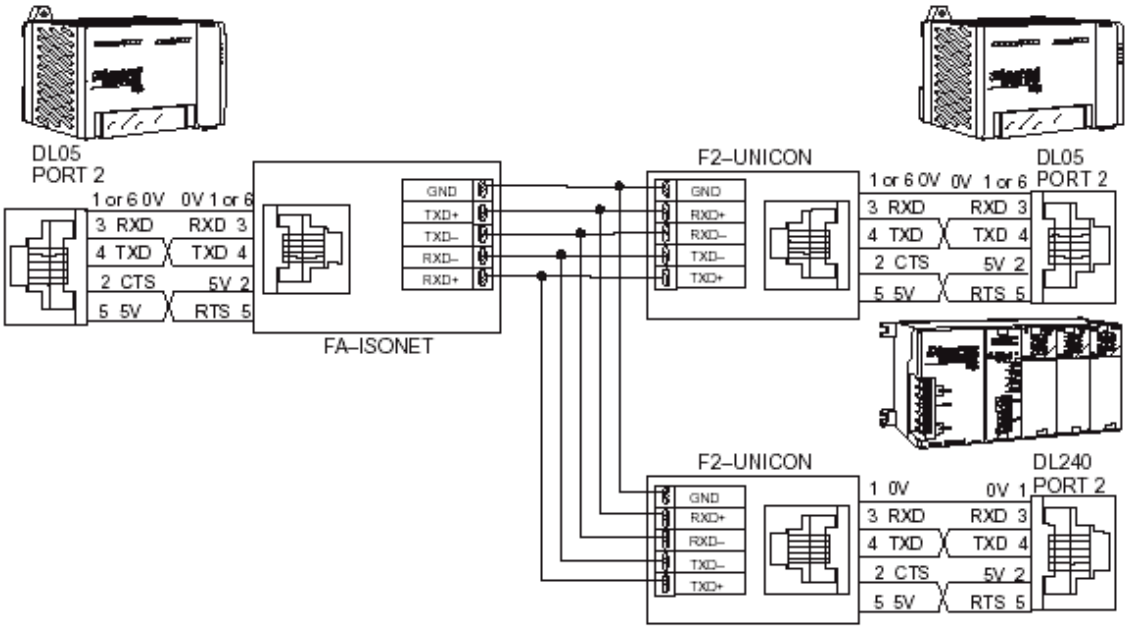


DL05 一对多连接（RS422）

DL05 与上位计算机的连接



DL05 作为主机连接其他 PLC



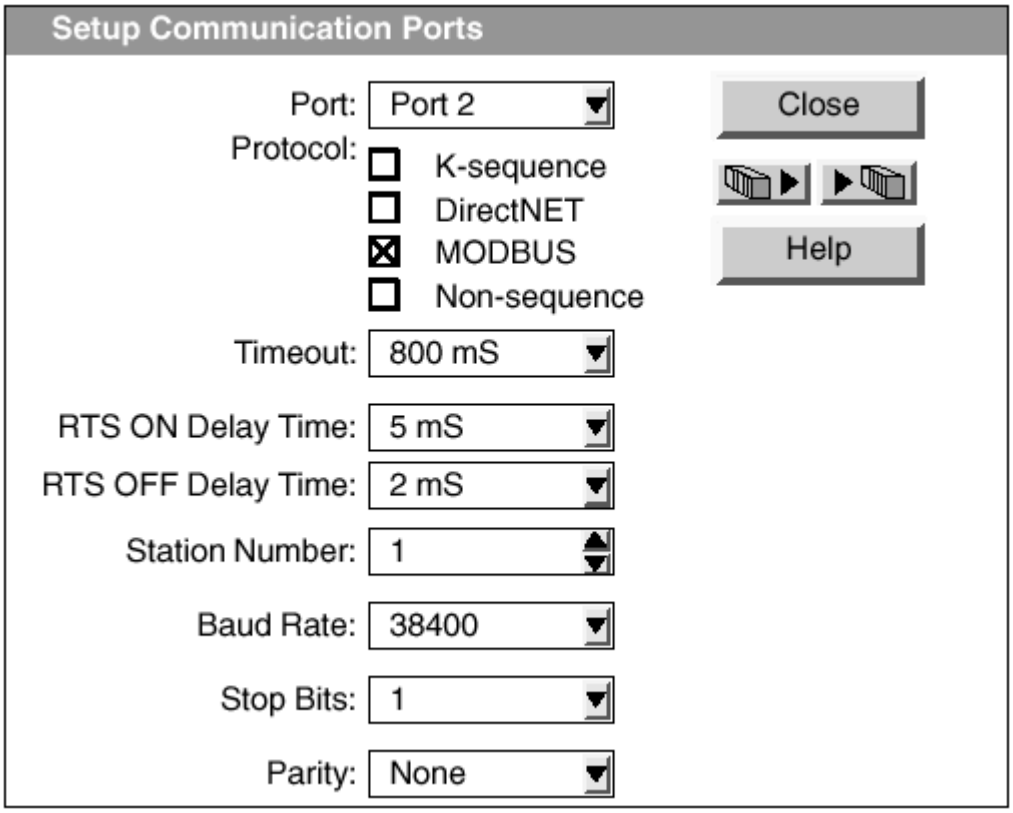
当把DL05加入多机通讯系统中时，RTS ON延时时间必须设置成5ms以上；RTS OFF 延时时间必须设置成2ms以上。必要时，可延长这2个时间。

MODBUS （CCM3）通讯时的通讯口设定

在 DirectSoft 软件中，选择‘ PLC ’菜单下‘ SETUP ’子菜单，选择“ Secondary Comm Port ”，然后进行各项选择，具体如下：

‘ Port ’：选择正确的通讯口，如图为通讯口 2；

‘ Protocol ’：选择协议为 MODBUS（在手持编程器上按 M56，选择‘ MODBUS ’）。



‘ Timeout ’：当通讯口发出信息后，等待回答的最长时间，如在该时间内没有接受到回答，则登记超时出错；

RTS ON/OFF 延时时间：RTS ON 延时定义当 DL05 置位 RTS 信号线后，发送数据前的等待时间；RTS OFF 延时定义当 DL05 发送完数据后，等待多长时间后复位 RTS 信号线。当 DL05 处于多机通讯系统中时，RTS ON 时间必须在 5ms 以上；RTS OFF 时间必须在 2ms 以上。

局号：当把通讯口作为 MODBUS 主局使用时，请选择‘ 1 ’。可用的 MODBUS 子局的范围是 1~247，但在主局方式下使用 DL05 通讯指令能存取的子局范围为 1~99。每个子局必须有唯一的局号。在上电时，通讯口自动置为子局，仅仅在 DL05 执行通讯指令时，通讯口才处于主局方式；此后，通讯口又返回子局方式，直到 DL05 再次执行通讯指令。

波特率：通讯速率，网络中的所有局的波特率必须相同。可用的波特率有：300，600，1200，2400，4800，9600，19200，38400（bps）。

停止位：选择 1 位或 2 位停止位。

奇偶校验：可选择无校验，偶校验，奇校验。

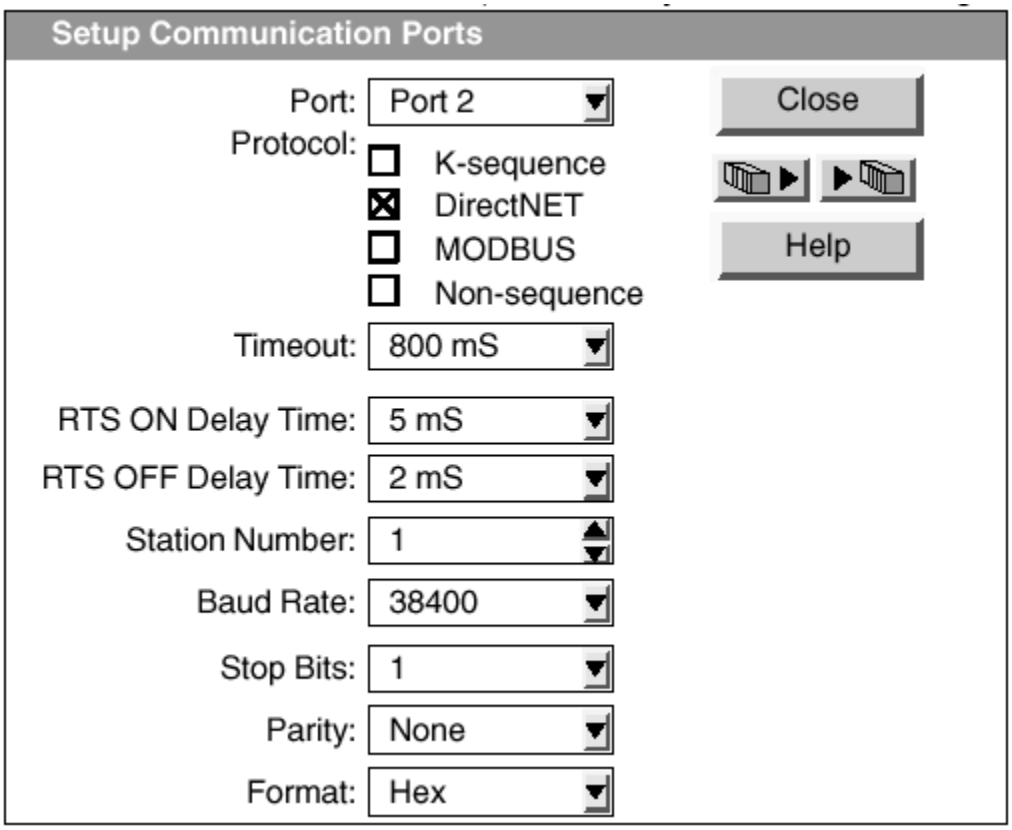
设置完成后，按该按键，把通讯口设置信息写入 PLC，然后按‘ CLOSE ’退出设置菜单。

CCM2（DirectNET） 通讯时的通讯口设定

在 DirectSoft 软件中，选择‘ PLC ’菜单下‘ SETUP ’子菜单，选择“ Secondary Comm Port ”，然后进行各项选择，具体如下：

‘ Port ’：选择正确的通讯口，如图为通讯口 2；

‘ Protocol ’：选择协议为 DirectNET（在手持编程器上按 M56，选择‘ DirectNET ’）。



‘ Timeout ’：当通讯口发出信息后，等待回答的最长时间，如在该时间内没有接受到回答，则登记超时出错；

RTS ON/OFF 延时时间：RTS ON 延时定义当 DL05 置位 RTS 信号线后，发送数据前的等待时间；RTS OFF 延时定义当 DL05 发送完数据后，等待多长时间后复位 RTS 信号线。当 DL05 处于多机通讯系统中时，RTS ON 时间必须在 5ms 以上；RTS OFF 时间必须在 2ms 以上。

局号：当把通讯口作为 CCM2 主局使用时，请选择‘ 1 ’。可用的 CCM2 子局的范围是 1~90，每个子局必须有唯一的局号。在上电时，通讯口自动置为子局，仅仅在 DL05 执行通讯指令时，通讯口才处于主局方式；此后，通讯口又返回子局方式，直到 DL05 再次执行通讯指令。

波特率：通讯速率，网络中的所有局的波特率必须相同。可用的波特率有：300，600，1200，2400，4800，9600，19200，38400（bps）。

停止位：选择 1 位或 2 位停止位。

奇偶校验：可选择无校验，奇校验。

数据格式：可选择 HEX 或 ASCII 码格式。

设置完成后，按该按键，把通讯口设置信息写入 PLC，然后按‘ CLOSE ’退出设置菜单。

网络子局工作方式

本节介绍网络中的其他设备如何与设置成 CCM2 或 MODBUS 子局的 DL05 进行通讯。

在进行 MODBUS 通讯时，MODBUS 通讯主局必须采用 MODBUS RTU 协议与 DL05 子局进行通讯，此时，主局必须发出 MODBUS 通讯功能号和对应于 PLC 功能号的 MODBUS 地址码。CCM2 通讯主局采用通常 I/O 地址方式存取 DL05 CPU 和系统。在以上的通讯中，不需要任何的 PLC 梯形图程序。

MODBUS 通讯功能码

MODBUS 功能码用于说明是读操作还是写操作，读写的内容是单个数据点还是一组数据。DL05 支持的 MODBUS 功能码如下表所示：

MODBUS 功能码	通讯功能	可用的 DL05 数据类型
01	读一组线圈	Q，M，T，C
02	读一组输入信号	I，SP
05	置位/复位单个线圈	Q，M，T，C
15	置位/复位一组线圈	Q，M，T，C
03，04	读单个或多个数据寄存器	R
06	改写单个数据寄存器	R
16	改写一组数据寄存器	R

MODBUS 地址的确定

典型的，有 2 种方法可用于确定 PLC 功能地址

1） 通过指定 MODBUS 数据类型和地址

有很多的 MODBUS 主局软件允许你通过指定 MODBUS 数据类型和地址的方式来确定 PLC 的功能地址，这是最容易的方法，但不是所有的软件都允许你用该种方式。

正确的计算地址的算式取决于你所选择的 PLC 数据的类型。在此种方式下，PLC 数据分成 2 大类：

离散型——I，SP，Q，M，S，T，C

连续字——R，定时器/计数器当前值

你首先要把 8 进制地址值转换为 10 进制数，然后加上适当的 MODBUS 地址偏移量（如果需要）。下表给出了对应于 DL05 所有功能存储器的 MODBUS 地址。

DL05 功能存储器	数量	地址范围（8 进数）	MODBUS 地址范围（10 进数）	MODBUS 数据类型
对于离散型数据：		10 进数表示的 PLC 地址 + 开始地址	+ 数据类型	
输入（I）	256	I0——I377	2048——2303	输入
特殊线圈（SP）	512	SP0——SP777	3072——3583	输入
输出（Q）	256	Q0——Q377	2048——2303	线圈
中间线圈（M）	512	M0——M777	3072——4583	线圈
定时器（T）	128	T0——T177	6144——6271	线圈
计数器（C）	128	C0——C177	6400——6527	线圈
级（S）	256	S0——S377	5120——5375	线圈
对于连续型数据		10 进数表示的 PLC 地址	+ 数据类型	
定时器当前值 R	128	R0——R177	0——127	输入寄存器
计数器当前值 R	128	R1000——R1177	512——639	输入寄存器
CMOS 寄存器 R	3968	R1200——R7377	640——3839	保持寄存器
不挥发寄存器 R	128	R7600——R7777	3968——4095	保持寄存器

下面给出几个该种方式下，如何取得 PLC 地址对应的 MODBUS 地址的例子。

1） R2100
A）在表中得到 R2100 对应的表项
B）把 R2100（O）转换为 10 进数 = 1088 （D）
C）加入表中对应的 MODBUS 数据类型
则，R2100 对应的 MODBUS 地址如下：
1088 + Hold.Reg = Holding Reg.1088

2） Q20
A）在表中得到 Q20 对应的表项
B）把 Q20（O）转换为 10 进数 =16（D）
C）加入表中对应的开始地址(2048)
D）加入表中对应的 MODBUS 数据类型
则，Q20 对应的 MODBUS 地址如下：
16 + 2048 + Coil = Coil.2064

3） T10 当前值
A）在表中得到 T10 对应的表项
B）把 T10（O）转换为 10 进数 = 8（D）
C）加入表中对应的 MODBUS 数据类型
则，T10 对应的 MODBUS 地址如下：
8 + Input.Reg = Input Reg. 8

4） M54
A）在表中得到 M54 对应的表项
B）把 M54（O）转换为 10 进数 =44（D）
C）加入表中对应的开始地址(3072)
D）加入表中对应的 MODBUS 数据类型
则，M54 对应的 MODBUS 地址如下：
44 + 3072 + Coil = Coil.3116

2） 仅通过指定 MODBUS 数据地址

有些 MODBUS 主局仅通过 MODBUS 数据地址来确定对应的 PLC 地址，这种方式下的地址指定方式有些不同，但这仍然是很简单的。基本的，该种方式下，仍然按地址范围划分数据类型，这意味着单凭地址就可精确描述数据类型和位置，一般的方法是给地址增加一个偏移量来实现。有一点非常重要，在这种方式下，你的上位主局软件可有 2 种编址方式：

- 484 方式
- 584/984 方式

我们强力推荐使用 584/984 编址方式。因该种方式下，能存取的地址空间较大。当你的主局软件仅支持 484 方式时，有些 PLC 地址有可能存取不到。正确的计算地址的算式取决于你所选择的 PLC 数据的类型。在此种方式下，PLC 数据分成 2 大类：

离散型——I，SP，Q，M，S，T，C
连续字——R，定时器/计数器当前值

你首先要把 8 进制地址值转换为 10 进制数，然后加上适当的 MODBUS 地址偏移量（如果需要）。下表给出了对应于 DL05 所有功能存储器的 MODBUS 地址。

DL05 功能存储器	数量	地址范围（8 进数）	MODBUS 地址范围（10 进数）	484 方式地址	584/984 方式地址	MODBUS 数据类型
对于离散型数据：		10 进数表示的 PLC 地址 + 开始地址		+	方式地址	
输入（I）	256	I0——I377	2048——2303	1001	10001	输入
特殊线圈（SP）	512	SP0——SP777	3072——3583	1001	10001	输入
输出（Q）	256	Q0——Q377	2048——2303	1	1	线圈
中间线圈（M）	512	M0——M777	3072——4583	1	1	线圈
定时器（T）	128	T0——T177	6144——6271	1	1	线圈
计数器（C）	128	C0——C177	6400——6527	1	1	线圈
级（S）	256	S0——S377	5120——5375	1	1	线圈
对于连续型数据 ：		10 进数表示的 PLC 地址		+	方式地址	
定时器当前值 R	128	R0——R177	0——127	3001	30001	输入寄存器
计数器当前值 R	128	R1000——R1177	512——639	3001	30001	输入寄存器
CMOS 寄存器 R	3200	R1400——R7377	768——3839	4001	40001	保持寄存器
不挥发寄存器 R	128	R7400——R7577	3840——3967	4001	40001	保持寄存器
系统寄存器 R	256	R7600——R7777	3968——4095	4001	40001	保持寄存器

下面给出该种方式下，如何取得 PLC 地址对应的 MODBUS 地址的几个例子。

1） 2100 （584/984 方式）
A） 表中得到 R2100 对应的表项
B） 把 R2100（O）转换为 10 进数 = 1088 （D）
C） 加入该方式下对应的 MODBUS 方式地址（40001）
则，R2100 对应的 MODBUS 地址如下：
1088 + 40001 = 41089

2） Q20（584/984 方式）
A） 在表中得到 Q20 对应的表项
B） 把 Q20（O）转换为 10 进数 =16（D）
C） 加入该方式下对应的开始地址(2048)
D） 加入对应的 MODBUS 方式地址（1）
则，Q20 对应的 MODBUS 地址如下：
16 + 2048 + 1 = 2065

3）T10 当前值（484 方式）
A） 在表中得到 T10 对应的表项
B） 把 T10（O）转换为 10 进数 = 8（D）
C） 加入对应的 MODBUS 方式地址（3001）
则，T10 对应的 MODBUS 地址如下：
8 + 3001 = 3009

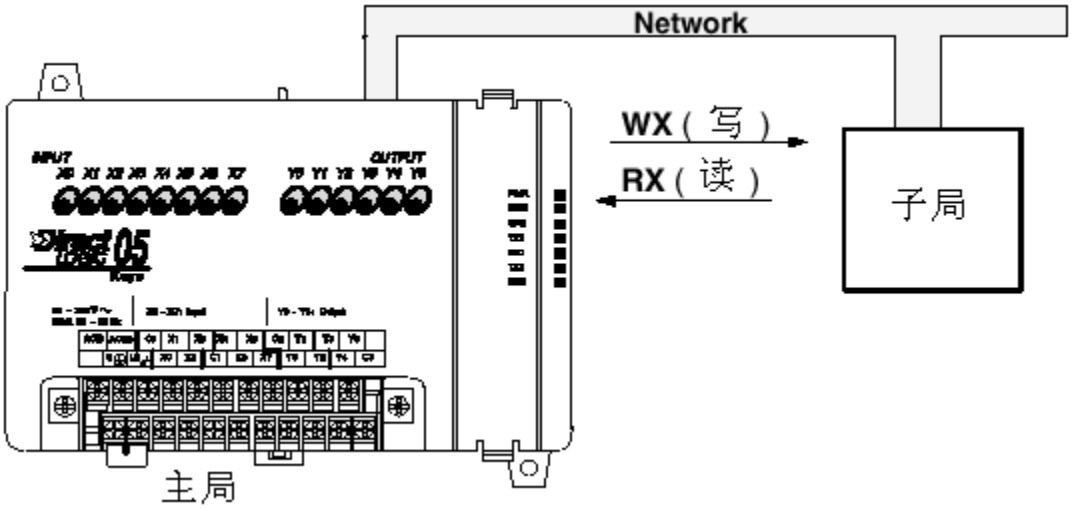
4） M54（584/984 方式）
A） 在表中得到 M54 对应的表项
B） 把 M54（O）转换为 10 进数 =44（D）
C） 加入对应的开始地址(3072)
D） 加入对应的 MODBUS 方式地址（1）
则，M54 对应的 MODBUS 地址如下：
44 + 3072 + 1 = 3117

CCM2 通讯地址的确定

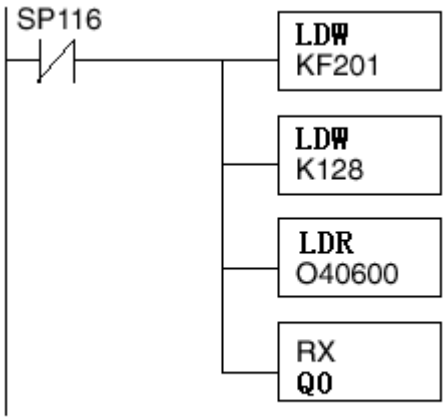
CCM2 通讯中子局地址的编址很简单，你只需直接采用子局的功能定义号。例如，要通过 CCM2 通讯存取子局的 R2000，你在主局上直接发出读取子局 R2000 的指令。

网络主局工作方式

本节介绍如何在 MODBUS，CCM2 通讯中把 DL05 作为主局来使用。MODBUS 和 CCM2 通讯都是单主局，多子局的主从式通讯网络。在该种网络中，仅有主局可发出通讯指令。本节介绍 DL05 作为网络主局使用时的逻辑梯形图指令。



DL05 仅需编制简单的通讯指令，即可实现通讯。往子局写通讯指令用 WX；从子局读通讯指令用 RX。在执行 WX/RX 通讯指令前，需在数据栈中设置好通讯所需要的相关参数，下面为通讯读数据指令的一个例子。



F：固定为 F
2：使用通讯口（BCD）
01：对方局号，MODBUS 最大 99，CCM2 最大 90

一次通讯数据字节数为 128
（最大 128）

本机存放通讯接受数据开始地址。
（为 PLC 功能地址号）

RX 为读指令，WX 为写指令
对方局开始地址为 Q0

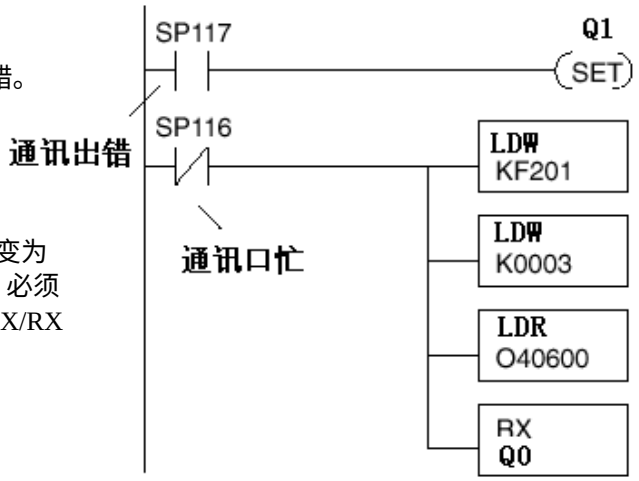
通讯对方子局的设定：

CCM2 子局：其子局号设置与 WX/RX 指令中子局号的一致；

MODBUS 子局：对于 SU/SZ/DL05/DL350 系列，其子局号设置与 WX/RX 指令中子局号的一致；

一般情况下，一次通讯的时间大于一个扫描周期。通讯口要等本次通讯结束后才能进行下一次通讯。对应于 DL05 的通讯口 2 有 2 个特殊线圈 SP116，SP117。SP116 为通讯口忙；SP117 为通讯口通讯出错。右边的例子中说明了在通讯程序中该特殊线圈的应用情况。在通讯进行中，SP116 一直为 ON，当 SP116 变为 OFF 后，可以进行下一次通讯。

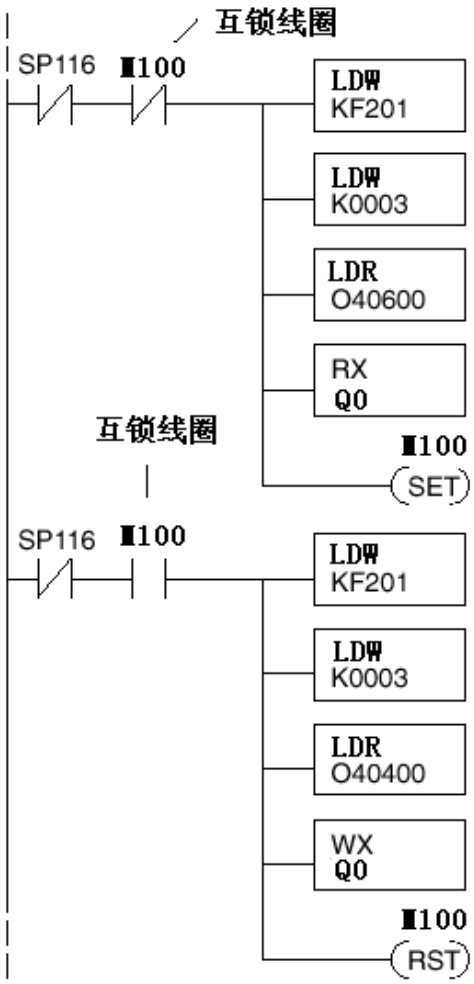
当 PLC 发现通讯出错时，置 SP117 变为 ON。当你需要在程序中使用该特殊线圈时，必须放在所有的通讯指令以前。因为每当执行 WX/RX 通讯指令时，PLC 会首先复位 SP117。



多重读写通讯的互锁

在有些情况下，你可能会在单个程序中执行多段通讯程序。为了保证每段程序都执行到，你必须对各段通讯程序采取互锁的方式，否则，PLC 仅扫描执行排在最先的通讯程序段，因为，PLC 在一个时间仅能执行一条指令。

右边的例子中，用 M100 作为 2 段程序的互锁线圈，当执行完 RX 指令后，置位 M100；此后，执行 WX 指令，并复位 M100。



第4章 高速计数和脉冲输出功能

第一节 简介

内装运动控制功能

DL05 CPU 内带有高速计数功能。通过特殊寄存器设定，高速计数功能可作为加法计数器，定位控制器，脉冲捕捉器等来使用。

高速输入特点：

· 高速计数(最大 5kHz)。24 段预置值和中断子程序。加计数带复位。
· 编码器输入加减计数(5kHz 最大)。加减计数带复位。
· 高速中断输入，立即响应外部事件或者定时任务。
· 脉冲捕捉，监视 1 个输入点的窄脉冲，最小宽度 100μs(0.1ms)。
· 软件滤波输入(on 和 off 延迟可达 99ms)，确保输入信号的完整(这是输入 I0-I2 的默认工作方式)。

脉冲输出特点：

· 单轴可编程脉冲输出（最大 7KHz），包括阶梯形定位、中断型定位和速度控制三种控制方式。

高速 I/O(HSIO)功能的适用性

重要：请注意以下限制

· 仅 DC 输入型 DL05 系列具有高速输入功能。
· 仅 DC 输出型 DL05 系列具有脉冲输出功能。
· 同一时间仅能使用一种 HSIO 功能。不能同时使用高速输入和脉冲输出。

选择高速 I/O 模式

6 种工作方式

DL05 高速 I/O 具有 6 种工作方式如下表，采用何种方式取决于表最左列的模式选择。

模式	方式名称	方式特性
10	高速计数	单通道加计数，最大 5KHz，24 段预置值，带复位
20	A/B 相计数	A/ B 输入，最大 5KHz，加/减计数，带复位
30	脉冲输出	步进控制-脉冲和方向信号，可编程的动作控制，最大 7KHz
40	外部中断	基于输入变化或时间产生一个中断。
50	脉冲捕捉	捕捉所选输入点的窄脉冲。
60	滤波输入	用软件的方法去掉对应输入点的干扰窄脉冲。

在选择六种高速 I/O 方式中的一种时，在下表中列出的 I/O 点只能按所列的功能工作。如果一个输入点不被特别用于支持一种特殊方式，通常滤波输入就被作为缺省方式使用。同样地，除非选择了脉冲输出方式，否则，输出点以正常方式工作。

实际的 I/O 点用途					
模式	DC 输入点			DC 输出点	
	I0	I1	I2	Q0	Q1
高速计数	计数时钟	滤波输入	滤波输入或计数复位	普通输出	普通输出
A/B 相计数	A 相输入	B 相输入	滤波输入或计数复位	普通输出	普通输出
外部中断	中断输入	滤波输入	滤波输入	普通输出	普通输出
脉冲捕捉	脉冲输入	滤波输入	滤波输入	普通输出	普通输出
脉冲输出	滤波输入	滤波输入	滤波输入	脉冲或 CW 脉冲	方向或 CCW 脉冲
滤波输入	滤波输入	滤波输入	滤波输入	普通输出	普通输出

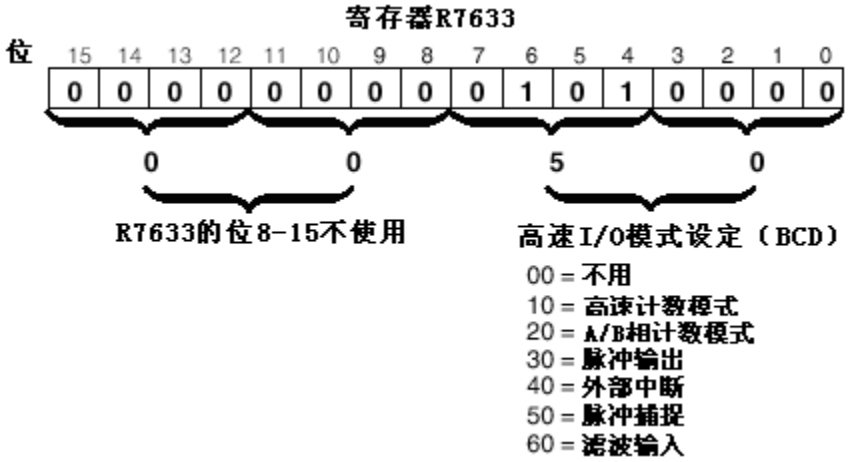
默认模式

默认模式是模式 60（滤波输入）。DL05 系列出厂时初始化为该模式，并在每次复位 R 寄存器时恢复为该模式。默认模式下，I0-I2 是滤波输入(10mS 延时)，Q0-Q1 是常规输出。

高速计数方式的设置

如果你已经选择了一种适用于你的需要的高速 I/O 方式，那么接着我们就准备进行 PLC 的设置。可以随时通过改变寄存器 R7633 的值来选择相应的工作模式。

R7633 的构成如下所示。



在 R7633 的低位字节中输入模式设定码（BCD 码）来选择相应的主工作模式，高位字节不用。例子中所示的数据“ 2050 ”为选择了方式 50—脉冲捕捉（BCD=50）。

I0~I2 的特殊设置

针对不同的工作模式，除了设置主工作模式码外，你可能还需要设定 R7634（对应于 I0），R7635（对应于 I1），R7636（对应于 I2）的相关内容。具体内容请参见高速 I/O 各工作模式。

R - 寄存器

模式		
I0	R7633	xxxx
I1	R7634	xxxx
I2	R7635	xxxx
	R7636	xxxx

第二节　模式 10：高速计数

DL05 的 HSIO 电路包括一个高速计数器。来自外部源（I0）的一个脉冲序列在每个信号的上升沿触发计数器。计数器只能从 0 到 99999999 进行加法计数。计数器把当前计数值与你定义的 24 段预置值进行比较。在该种工作模式下，它使用计数器 C76 和 C77。

当在 R7633 的低位字节中置入 BCD 数‘ 10 ’时，HSIO 就允许加计数方式，此时，I0 为信号输入端，每当它从 OFF 变成 ON 时，进行加计数；I2 为高速计数复位端，当其 ON 时，使计数器清零；I1 为滤波输入点，可在用户程序中使用。

你可通过设置，取消 I2 的计数器复位功能而使 I2 成为普通的滤波输入点，此时，对计数器的复位需要通过用户程序编程完成。

模式 10 的设定

要设置成模式 10，需 R7633 = 0010（BCD），对 R7633 的设定可有以下 3 种方法：

- 在用户程序中编制相应的程序；
- 使用 DirectSoft 软件的存储器编辑器或数据查看项；
- 使用手持编程器的相关功能。

模式 10 下 I0~I2 的特殊设定

在模式 10 下，输入点 **I0** 专门用于计数器信号输入。输入点 **I1** 可以是一个普通或滤波输入点。输入点 **I2** 可被设置成带/不带中断选择的计数器复位，中断选择允许复位输入点（**I2**）产生一个中断，就象预置值所做的那样，但是 **SP** 继电器不闭合（替代的，**I2** 将在中断子程序中接通 **1 个**扫描周期）。另外 **I2** 可能只简单地用作一个滤波输入点。

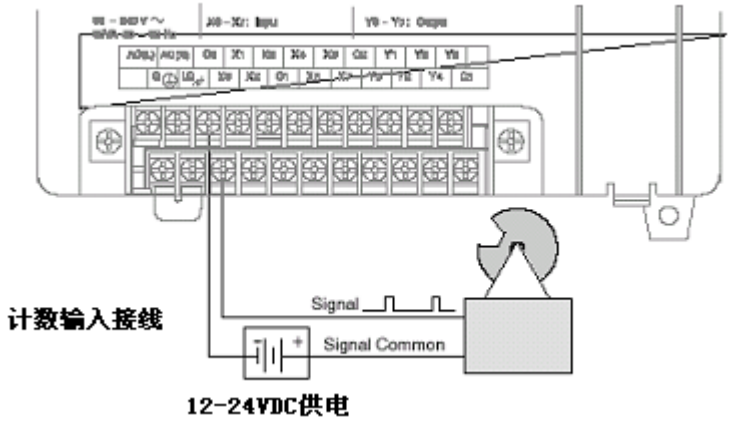
模式 10 下，I0~I2 的特殊设置如下表

输入	配置寄存器	功能	需要的十六进制码
I0	R7634	计数器输入	0001
I1	R7635	滤波输入	××06, ××=滤波时间 0-99ms(BCD)
I2	R7636	计数器复位（无中断）	0007*（缺省） 0207*
		计数器复位（有中断）	0107* 0307*
		滤波输入	××06, ××=滤波时间 0-99ms(BCD)

*带计数器复位时，你可以选择正常复位或快速复位。但是，在程序执行期间，快速复位不识别已改变的预置值。当在 **R7636** 中设定了‘ **0007** ’或‘ **0107** ’，并且在程序执行期间预置值被改变了时，**DL05** 会在复位时识别到已改变的预置值。当在 **R7636** 中设定了‘ **0207** ’或‘ **0307** ’时，**CPU** 不检查已改变的预置值，从而 **DL05** 就有了更快的复位时间。

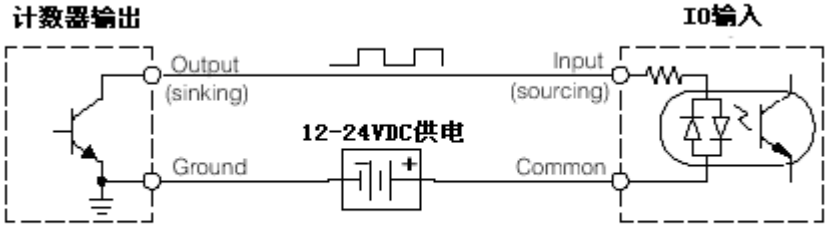
接线图

在**模式 10** 下，**DL05** 高速计数端与现场设备的一般接线图如下所示。它可以使用多种类型的脉冲发生器件，例如接近开关、单通道编码器，磁性或光传感器等等。带汇点输出的设备（**NPN** 集电极开路）是连接的最佳选择。如果计数器是源点输入，它必须输出 **12-24VDC**。注意，带 **5V** 源点输出的器件不能与 **DL05** 的输入端一起工作。

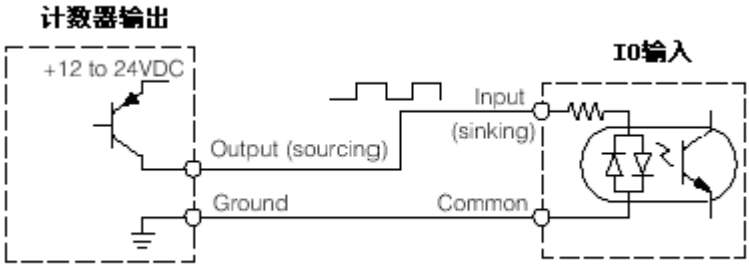


与计数器输出端的连接

DL05 的 **DC** 输入端的灵活性在于它们可检测任一方向的电流流动，所以它们可以被连接到源点输出或汇点输出的计数器输出端。在下列电路中，一个计数器有集电极开路 **NPN** 晶体管输出端。它从 **PLC** 输入点汇流。电源可以是 **FA-24PS** 或其它电源（**+12VDC** 或**+24VDC**），只要符合输入技术参数。



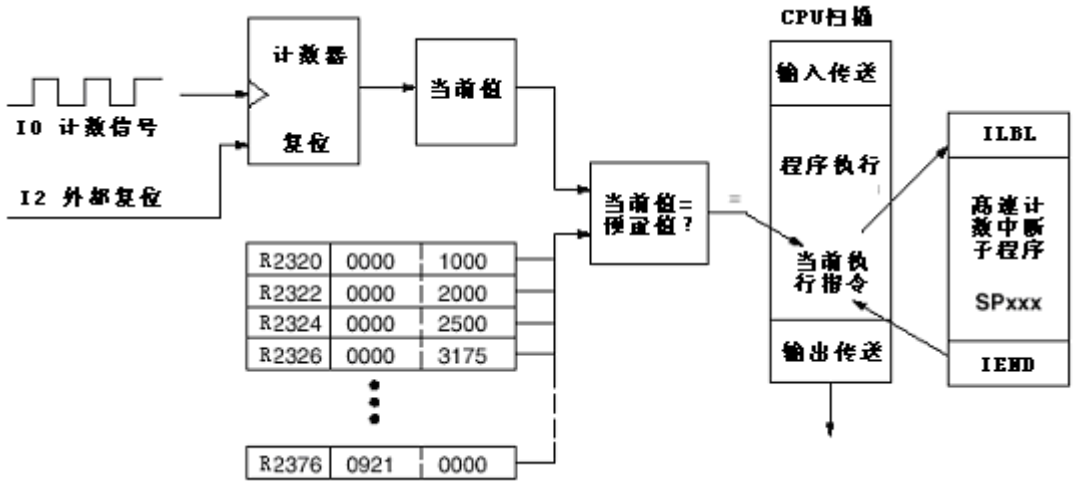
在下面的电路里，一个编码器有发射集开路 **PNP** 晶体管输出端。它作为 **PLC** 输入点的电流源，将电流汇至零电平。因为编码器提供电流源，所以不需另外加电源。但是，注意，编码器的输出必须是 **12-24V**（**5V** 的编码器输出端将无法工作）。



预置值和特殊继电器

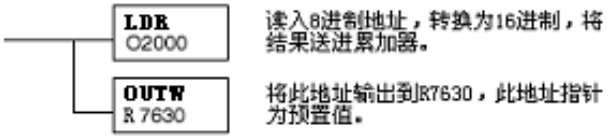
计数的目的是为了当计数值达到预置值时，进行一项特殊的工作。参见下图。DL05 高速计数器的特点是有 **24** 段预置值，你可以用编程的方法设置它们。预置值是一个你设置并保存起来的数字，计数器将不断地把当前计数值与预置值进行比较。当两值相等时，对应的特殊继电

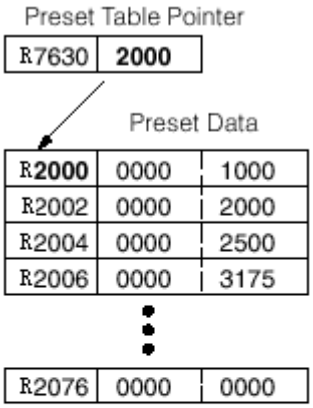
器触点接通，执行程序跳至中断程序。我们建议在中断服务程序中使用特殊继电器来完成你需要的直接动作。在中断服务程序结束后，**CPU** 返回到梯形图主程序，并继续从中断处执行主程序。此时比较功能为下一次预置中断事件作好了准备。



预置值数据的起始地址

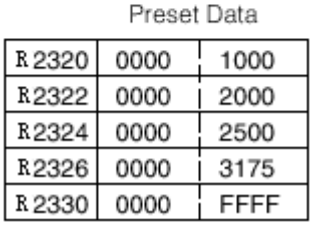
R7630 是一个指针地址，它指向预置值数据表的开头。预置值数据表缺省的起始地址是 **R2320**（把 **R 寄存器**初始化后该值为缺省值）。但是，你可以通过编程方法在 **R7630** 中放入不同的值来改变它。如下所示用 **LDR** 和 **OUTW** 指令：





使用少于 24 段预置值时

当使用的预置值寄存器少于 **24** 段时，请在紧随在最后的预置值寄存器后的寄存器中设置值“ **0000FFFF** ”，以表明已经到了最终设定值。右边的例子使用了四段预置值。**R2331-R2330** 中的 **0000FFFF** 表明了前一个预置值是最终设定值。



注意：每一个紧接其后的设定值都必须比它前一个设定值大。如果一个设定值比它的前一个设定值小时，**CPU** 就无法比较此值，因为计数器只能加法计数。

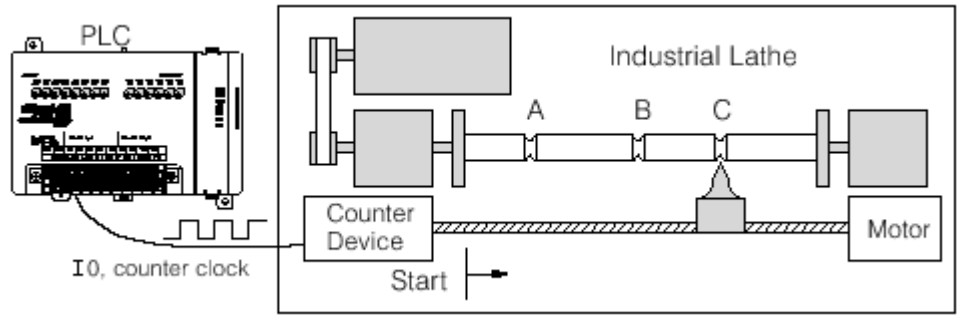
预置值一致继电器

下表列出了所有 **24** 段预置值寄存器的缺省值。每个寄存器占用两个 **16**-位 **R** 寄存器。相对应的继电器线圈号列在第二列中。我们也可称之为“一致”继电器，因为当高速计数器的当前值等于预置值时，它们的成立条件为真（闭合的）。“一致”继电器的触点会保持闭合，直到计数器当前值等于下一个预置值。

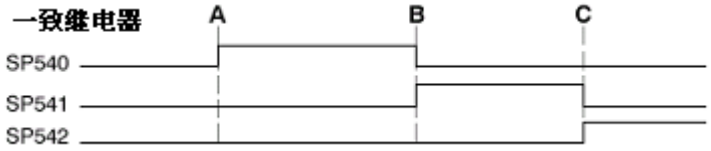
预置值	预置值 R 寄存器	特殊 继电器号	预置值	预置值 R 寄存器	特殊 继电器号
1	R2321/R2320	SP540	13	R2351/R2350	SP554
2	R2323/R2322	SP541	14	R2353/R2352	SP555
3	R2325/R2324	SP542	15	R2355/R2354	SP556
4	R2327/R2326	SP543	16	R2357/R2356	SP557
5	R2331/R2330	SP544	17	R2361/R2360	SP560
6	R2333/R2332	SP545	18	R2363/R2362	SP561
7	R2335/R2334	SP546	19	R2365/R2364	SP562
8	R2337/R2336	SP547	20	R2367/R2366	SP563
9	R2341/R2340	SP550	21	R2371/R2370	SP564
10	R2343/R2342	SP551	22	R2373/R2372	SP565
11	R2345/R2344	SP552	23	R2375/R2374	SP566
12	R2347/R2346	SP553	24	R2377/R2376	SP567

预置值的计算

预置值占用两个数据字。它们的数值范围可从 **0000 0000 到 9999 9999**，就和高速计数器值一样。所有 **24** 段预置值都是***绝对值型***数值，即每个值都是相对于计数器零值的一个偏移量。预置值必须根据每次的使用单独地取出来。在下面的工业车床原理图中，**PLC** 通过计脉冲数来监控螺杆的位置。在沿轴向的 **A**，**B** 和 **C** 点处，刀具头切入加工材料并切割出一条槽。



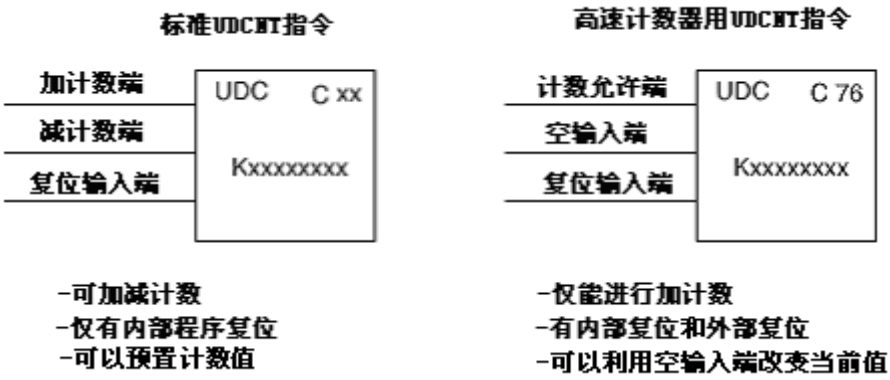
下面的时序图说明了对应的每个一致继电器闭合的周期。各自的继电器将保持 **on**，直到下一个继电器闭合。当计数器复位时，所有的继电器均变成 **off**。



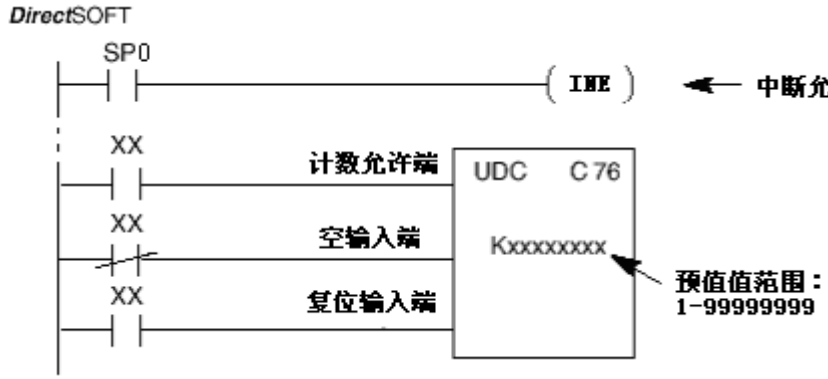
注意：每个紧接其后的预置值必须是比前一个预置值大 2 的数字。在工业车床的例子中，**B>A+1** 且 **C>B+1**。

高速计数的控制程序

高速计数器的助记符为 **UDCNT**（加/减计数器），在 **DL05** 指令系统中 **UDCNT** 计数器指令是一个标准指令，2 者比较参见下图。**DL05** 有 **128** 个计数器，记为 **C0-C177**。在模式 10 下，高速计数器使用 **UDC C76**（实际使用 **C76** 和 **C77**）。它有三个输入端，如下所示。第一个输入端（**Enable**）用于计数允许。中间的输入端是空端子，可以利用它来改变计数器当前值。最后一个信号是复位端。在预置当前值期间，计数允许端信号必须为 **OFF**。当计数器进行计数时，空输入端必须为 **OFF**。



下面的图说明了在梯形图程序中高速计数是如何表示的。注意，中断允许（**INE**）指令必须在计数器值达到第一个预置值之前被执行。通过使用一次扫描继电器 **SP0**，我们可以在上电时执行此操作。当使用高速计数器而不用预置值和中断子程序时，我们可以省略 **INE**。

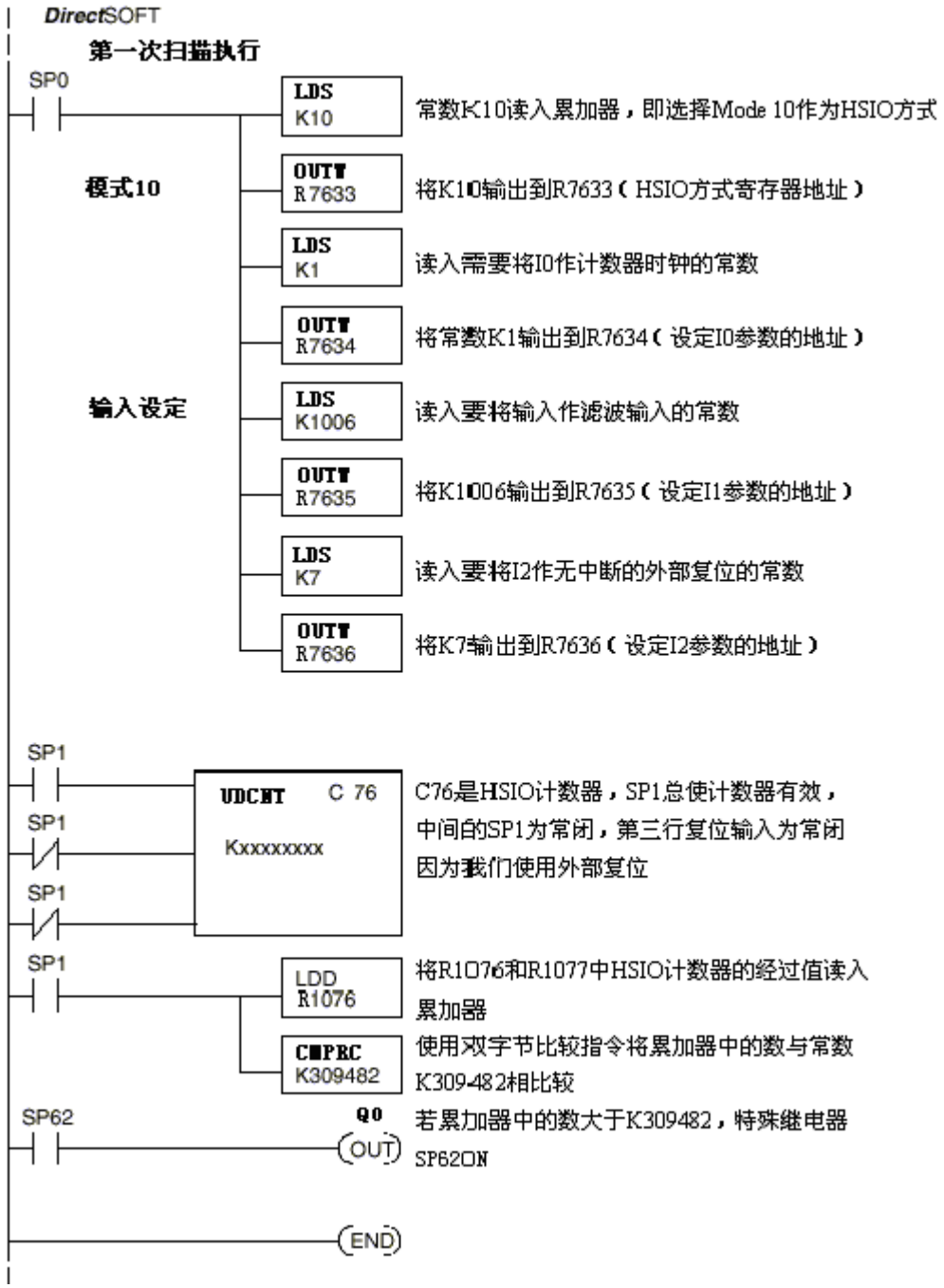


当允许计数输入端被激励时，高速计数器响应 **I0** 上的脉冲并且计数器 **C76-C77** 的计数值增加。外部硬复位输入端 **I2**（当你已选用它时）和程序中的复位信号以逻辑或的方式作为复位输入触点。所以，高速计数器收到的复位信号可以来自梯形图复位回路中的触点，或者，如果你已经把 **I2** 设定成一个外部复位端了，那么它也可以来自外部复位端 **I2**。

程序举例：无预置值的计数器

下面的例子是使用高速计数器的最简单的方法，在中断程序中它没有用预置值和特殊继电器。程序按方式 **10** 来设置高速计数电路，所以 **I0** 自动成为计数器信号输入端。它使用 32 位比较（**CMPRD/CMPRC**）指令，从而在某一计数值下产生动作。注意，在此方式下，允许你使用 **24** 段以上的“ 预置值 ”。另外，它把 **I2** 设置成计数器的外部复位端。

程序例子如下：



上面的程序中，使用 **32** 位比较指令来比较高速计数器当前计数值与某个常数，从而来接通 **Q0**。此方法产生的比较可能多于 **24** 次，但它与扫描时间有关，比较多了，其扫描时间就长，对计数的反应就慢。如果你需要使用非常快的响应时间，那么就要用内置的 **24** 段预置值和中断程序，如下面的例子所示。

带预置值的计数器的程序举例

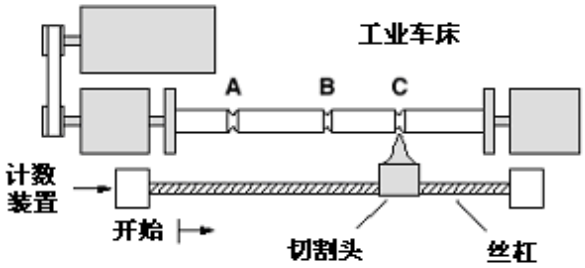
下面的例子说明如何用三个预置值来编写高速计数程序。你可能还记得本章开头的工业车床的例子。该程序例子说明了如何控制刀具头在工件的精确位置加工出三条槽。当螺杆转动时，计数装置将产生 **DL05** 能计数的脉冲。三个预置值变量 **A，B** 和 **C** 代表了这三条槽各自的对应位置（脉冲数）。

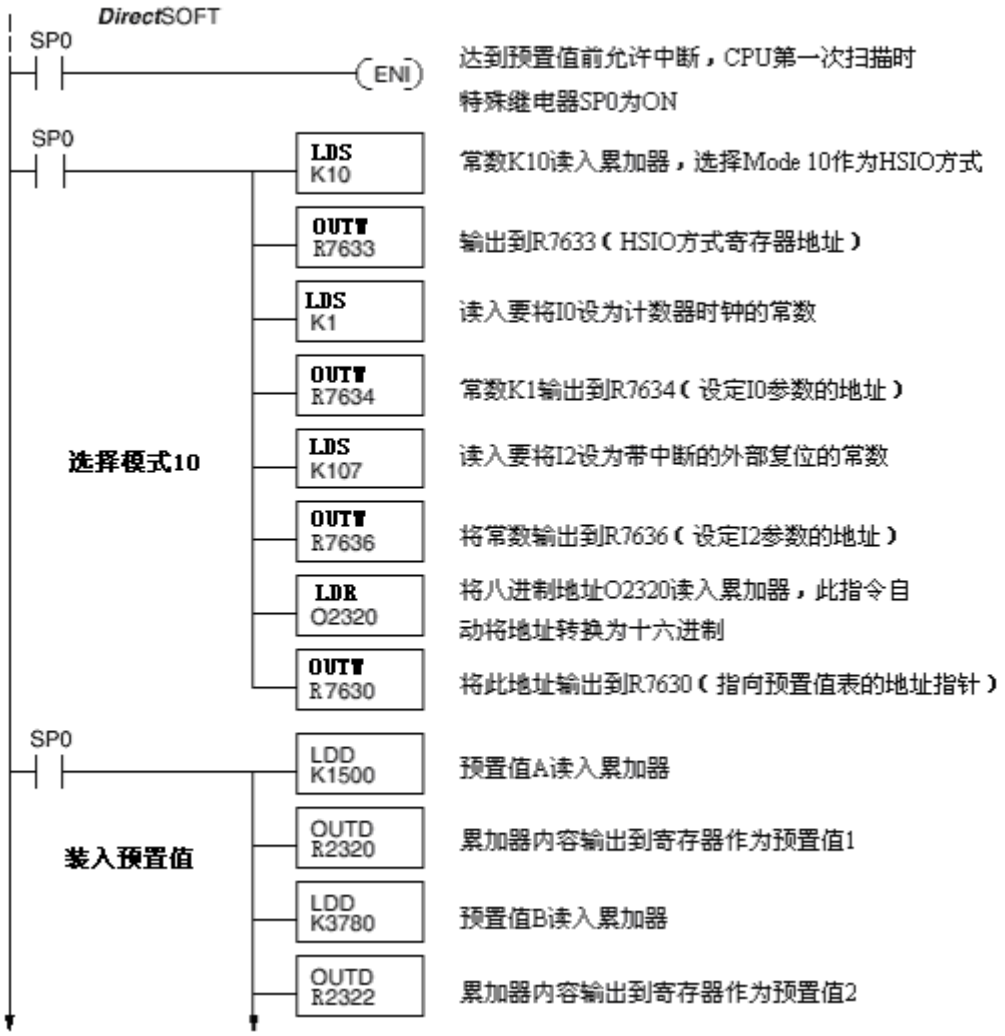
预置值

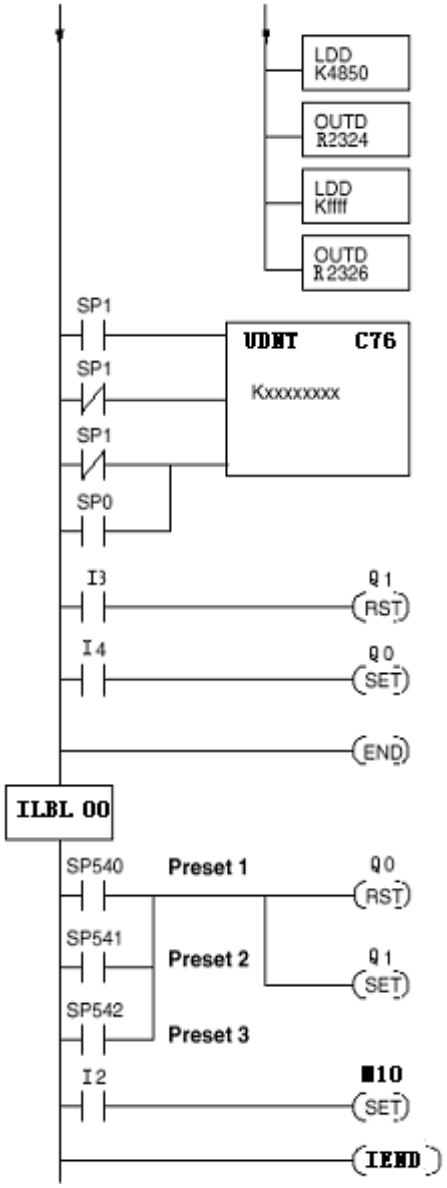
A	R2320	0000	1500
B	R2322	0000	3780
C	R2324	0000	4850
	R2326	0000	FFFF

I/O 分配

I3-切割到位信号
I4-切割头复位信号
Q0-丝杠旋转马达
Q1-切割头线圈控制







预置值 C 读入累加器
累加器内容输出到寄存器作为预置值 3

常数 Kffff 读入累加器，此值表示预置值列表结束

累加器内容输出到寄存器作为预置值 4（预置结束标志）

C76 是高速计数器，第一行的 SP1 为允许计数
中间的空输入端为 off（不使用）
第三行的复位输入是常闭，因为我们使用外部复位，
在每次上电时可以使用 SP0 触点将计数器复位。

当工件槽口切割完，I3 通电，切割头缩回

切割头缩回后，导引丝杠再次通电动作

主程序结束

中断服务程序开始

立即关断导引丝杠电机

当达到各个预置值时，这些“ 一致 ”继电器各自闭合，
每个结果都会使 Q1 输出，切割出一个槽，因此它们是
逻辑或的关系。

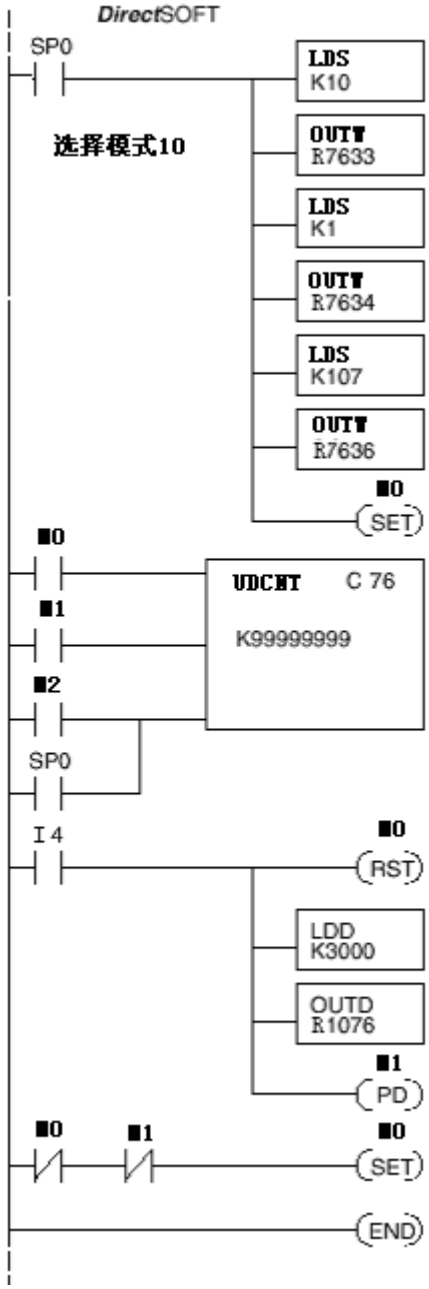
如果 I2 是外部中断源，则中断程序中的 I2 是得电的

中断服务程序结束

某些应用场合需要在各个预置值下有不同的动作。通过接通唯一对应的继电器线圈 **SP**×××，中断程序可以把一个预置事件与另一个区分。通过一个个地检查一致继电器线圈，我们可以确定出中断源。对于 I2，如果外部复位输入端—**I2** 产生了中断（如果设置成 I2 可以产生中断的话），那么触点 I2 将接通（仅在中断程序中）。

计数中改变计数器当前值的程序例子

下面的例子说明你如何把一个值读入预装为当前计数值。当预先读入信号的输入端（此例中为 **I4**）得电接通时，计数允许信号 **M0** 失效。接着我们把数值 **K3000** 写入计数寄存器（**R1076-R1077**）。我们预先将 **K3000** 读入作为计数器的当前值。当预先读入信号（**I4**）失效时，计数器继续计脉冲数，但现在是从 **K3000** 开始计数。



K10 读入累加器，即选择 Mode 10 作高速计数方式
输出到 R7633 寄存器（高速计数器方式寄存器地址）

读入要将 I0 作计数器计数允许常数
K1 输出到 R7634（设置 I0 参数的地址）

读入要将 I2 作带中断的外部复位的常数
将常数输出到 R7636，（设置 I2 参数的地址）

上电时 M0 置为 ON，允许计数

C76 是高速计数器，第一行的 M0 触点允许计数，中间行是空输入端
第三行的复位输入端是常闭的，因为我们使用外部复位，在每次上电时使用 SP0 接点将计数器复位

I4 ON，首先复位 M0

将 BCD 值 K3000 读入累加器
将常数输出到 R1076/R1077（C76 累加计数值地址）

产生一个初始化计数输入脉冲，将 R1076-R1077 初始化

初始化后（M1 OFF），将 M0 置位，允许计数

主程序结束

Mode 10 的故障检修指导

如果你使用模式 10 时遇到了问题，请仔细阅读下列故障现象和可能的原因。最常见的问题被列出如下。

- **故障症状：计数器不计数。**

可能的原因：

1．**现场传感器和接线**—检查核实编码器，接近开关，或计数器已接通并且 **I0** 的状态指示灯也发亮。那么问题可能是因为汇点-源点接线问题，等等。记得检查信号的地线连接。还要核查脉冲的接通持续时间是否足够长，以使 **PLC** 能识别它。

2．**配置**—用数据视窗检查配置参数。**R7633** 必须被置为 **10**，**R7634** 必须被置为 **1**，使高速计数方式有效。

3．**保持在复位状态**—检查复位输入端 **I2** 的输入状态。如果 **I2** 为 **ON**，因为计数器被保持在复位状态，所以无法计数。

4．**梯形图程序**—保证在你的程序中使用计数器 **C76**。上面的输入端是计数器的允许信号。在计数器计数前，它必须为 **ON**。中间的输入端是空端子。最后一个输入端是计数器复位，在计数时它必须为 **OFF**。

- **故障症状：计数器计数但预置值不起作用。**

可能的原因：

1．**配置**—保证预置值是正确的。预置值是 **32**-位 **BCD** 值，范围从 **0** 到 **99999999**。你一定要用 **LDD** 和 **OUTD** 指令把 **32** 位全部写入预定的地址中。只能用从 R2320 到 R2376 的整数地址。如果使用的预置值不到 **24** 段，在被使用的最后一个预置值后面的地址中一定要放入“**0000FFF**F”。

2．**中断程序**—仅用中断**#0**。在需要中断之前先执行 **INE** 指令，就保证了中断的有效工作。中断程序必须被放在主程序的后面，用 **ILBL** 标出并以中断返回指令 **IEND** 结束。

3．**特殊继电器**—检查你程序中的特殊继电器号。预置值 **1** 使用 **SP540**，预置值 **2** 使用 **SP541**，等等。记住，一次只能接通一个特殊一致继电器。当计数器值达到下一个预置值时，已接通的 **SP** 触点现在断开，下一个触点则接通。

- **故障现象：计数器加法计数，但无法复位。**

可能原因：

当你想要复位时，检查 **I2** 的状态指示灯，肯定它是工作的。或者，如果你正在使用一个内部复位端的话，用 ***Dire*CSOFT** 的状态方式或手持编程器监控计数器的复位输入端。

第三节　模式 20：A/B 相计数

HSIO 电路中的计数器可以对两个 **90** 度相位差的单脉冲，而不是一个单脉冲序列（工作方式 **10**）进行计数。**A/B** 相信号通常产生于增量型的旋转编码器或线性编码器。**A/B** 相计数器有两种范围，从 **0-99999999** 或**-8388608-8388607**。用 **C76** 和 **C77**，**A/B** 相计数器可以在最大 **5KHz** 频率下计数。与 **Mode 10** 不同的是，**Mode 20** 可以加/减计数，但是不具备 24 段预置值或“靠外部复位产生中断”的能力。它同样使用计数器 **C76**，**C77**。

模式 20 的设定

在 **R7633 的**低字节中写入 **BCD** 数 **20** 就选择了 A/B 相高速计数方式。**DL05** 不使用 **R7633** 的高位字节。

设置 **R7633** 的办法有以下几种：

- 在你的梯形图程序中编入 **LDW** 和 **OUTW** 指令。
- 利用 ***Direct*Soft** 软件的存储器编辑器。
- 用手持编程器直接修改寄存器值。

我们推荐使用上面的第一种方法以便设置的 **HSIO** 成为你应用程序的一个完整部分。在该段落后面的程序例子说明了如何应用它。

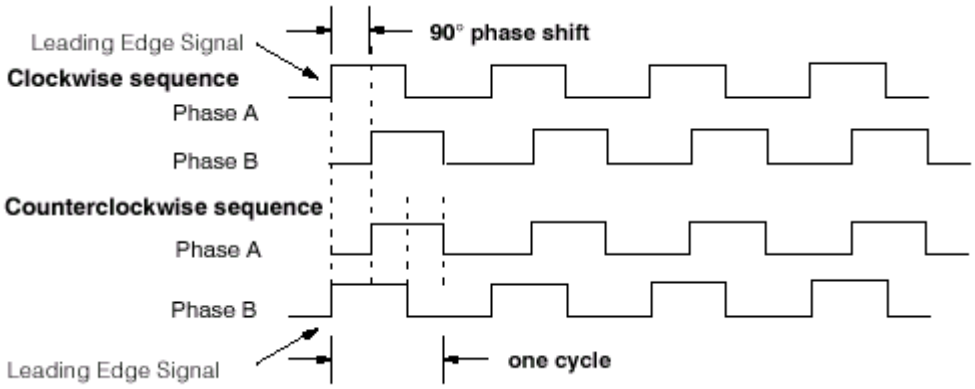
I0~I2 的特殊设定

下表列出了在模式 20 下 I0~I2 的可设定内容。输入点 **I0** 专门用于 **A** 相输入，输入点 **I1** 用于 **B** 相输入。输入点 **I2** 是复位输入端，但它不能产生中断。

输入端	配置寄存器	功能	需要的十六进制码
I0	R7634	A 相	0002（缺省） 90 度相位差，绝对值 0-99999999
			0012 90 度相位差，绝对值 -8388608-8388607
I1	R7635	B 相	0000
I2	R7636	计数器复位（无中断）	0007
		独立的滤波输入	1006

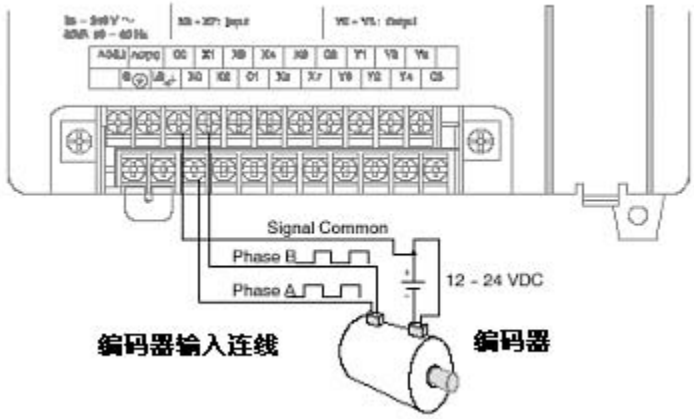
A/B 相编码器信号

A/B 相编码器信号包括了位置和方向信息，而它们的频率代表了移动的速度。下面所示的 **A** 相和 **B** 相信号为 **90** 度相位差，所以称为正交的相位差。当 **A** 相的上升沿领先于 **B** 相的上升沿时（习惯上表示是顺时针运转），**HSIO** 计数器就做加法计数。如果 **B** 相的上升沿领先于 **A** 相的上升沿（表明是逆时针运转），计数器就做减法计数。



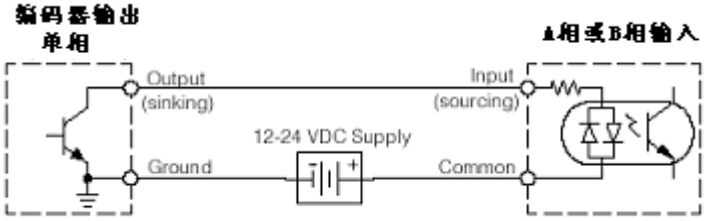
接线图

在 **HSIO 模式 20** 方式下，编码器和 **DL05** 的一般接线图如下所示。汇点输出（**NPN** 集电极开路）的编码器可能是连接的最好选择。如果编码器是源方式的，它必须输出 **12-24VDC**。注意，带 **5V** 电源输出的编码器不能与 **DL05** 输入端一起工作。

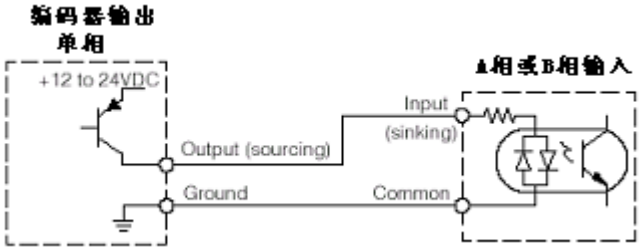


与编码器输出端的连接

DL05 的 **DC** 输入端的灵活性在于它们可检测任一方向的电流流动，所以它们可以被连接到编码器的源点输出端或汇点输出端。在下列电路中，一个编码器有集电极开路 **NPN** 晶体管输出端。它汇集来自 **PLC** 输入点的电流，即电流源。电源可以是**+24VDC** 辅助电源或其它电源（**+12VDC** 或**+24VDC**），只要符合输入技术参数。

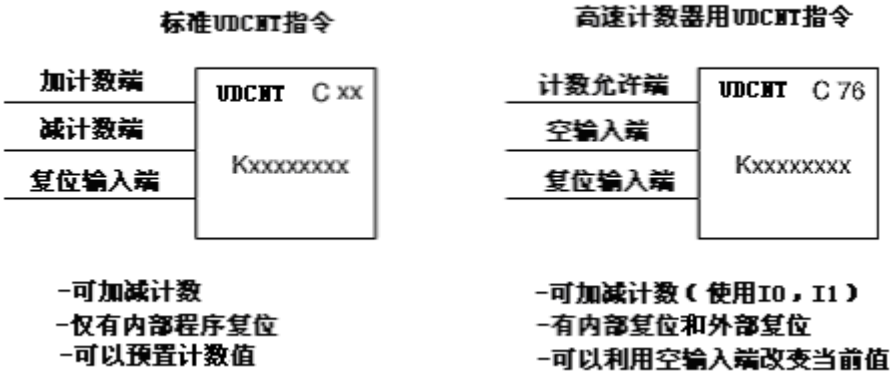


在下面的电路里，一个编码器有发射极开路 **PNP** 晶体管输出端。它作为 **PLC** 输入点的电流源，电流汇至零电平。因为编码器提供电流源，所以不需另外加电源。但是，注意，编码器的输出必须是 **12-24V**（**5V** 的编码器输出端将无法工作）。

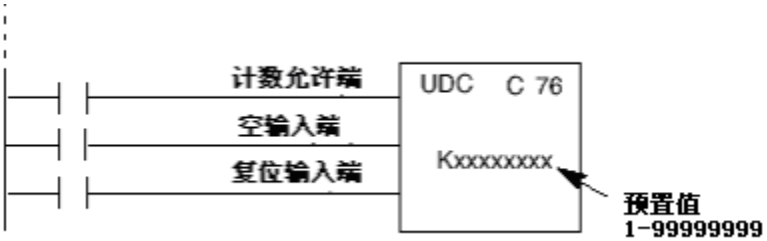


A/B 相计数的控制程序

用 **UDCNT C76**，你可以在梯形逻辑图中使用 **HSIO** 电路中的 **A/B** 相计数器。因为 **HSIO** 计数器是一个双字节计数器，所以它需要两个计数器地址。当 **HSIO** 模式 **20** 有效工作时，计数器 **C76** 和 **C77** 被用作 **A/B** 相计数器。与标准的 **UDCNT** 指令一样，它也有三个输入端，如下图所示，但它们被重新定义了。第一个输入端（**Enable**）是计数允许端，中间的输入端是预置端子（写入），最后一个端子是复位信号端。当计数器进行计数之前，计数允许端信号必须为 **ON**。在预置当前值期间，计数允许端信号必须为 **OFF**。



下面的图说明了在梯形图程序中 **HSIO A/B** 相计数器是如何表示的。

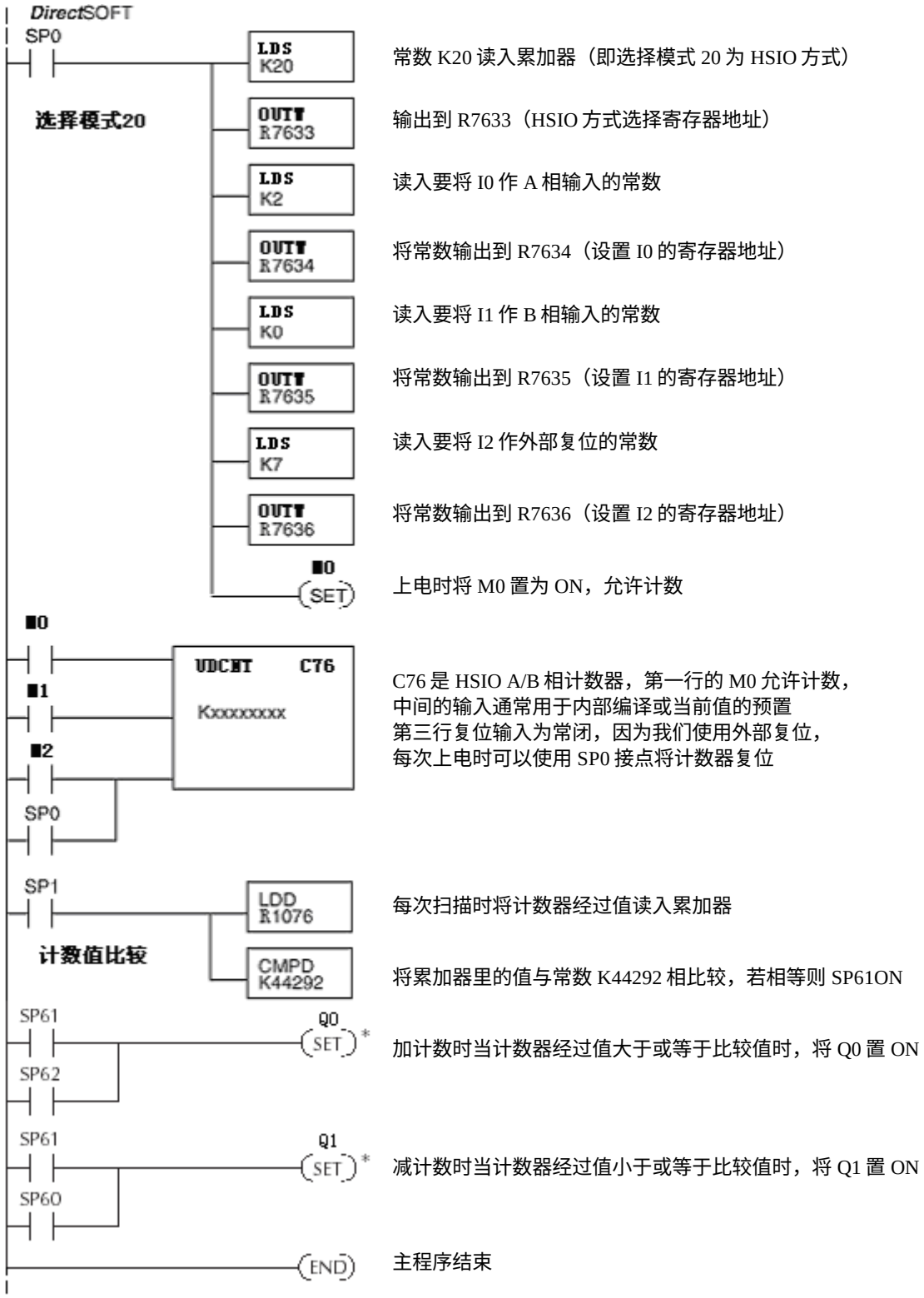


当计数允许端得电闭合时，计数器响应 **I0** 和 **I1** 上的 **A/B** 相脉冲，计数器 **C76-C77** 的计数值增加或减小。复位输入端的信号与外部的复位输入端 **I2** 以逻辑 **OR** 的方式作为计数器的复位输入条件。这说明 **A/B** 相计数器收到的复位信号可以来自梯形图复位回路中的触点上，或者，也可以来自外部复位端 **I2**。

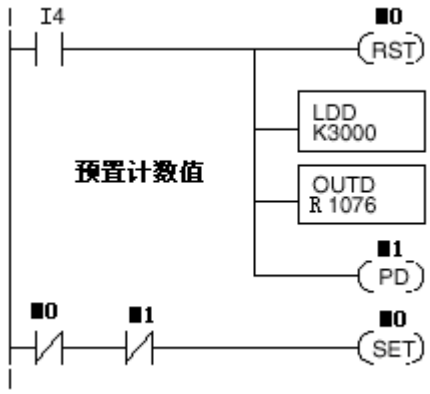
A/B 相计数器设置及预设计数器当前值的程序例子

因为预置值不适用于 **90** 度相位差计数，所以此方式最适用于简单的计数和测量。下一页的程序例子说明了如何配置 **A/B** 相计数器。程序把 **HSIO** 电路设置为 **Mode 20**，所以 **I0** 为 **A** 相，**I1** 为 **B** 相信号输入端。

程序例子如下：



要实现计数值预置功能，只要将下面的回路添加到上面的程序中。



当需要初始化时，将 I4 闭合，首先将 M0 复位

BCD 值 K3000 读入累加器

将常数输出到 R1076/R1077 即脉冲计数累加值地址
产生一个计数器初始化输入脉冲，将 R1076-R1077 内的计数值初始化

初始化后（M1 OFF），将 M0 置位，允许计数

模式 20 的故障检修指导

如果你使用模式 **20** 时遇到了问题，请仔细阅读下列故障现象和可能的原因。最常见的问题被列出如下：

- **故障症状：计数器不计数。**

可能的原因：

1．**现场传感器和接线**—检查核实编码器或其它电子设备的输入端已接通并且 **I0** 和 **I1** 的状态指示灯也发亮。当慢慢旋转标准增量编码器（1RPM）时，I0 和 I1 的发光指示灯交替变亮。那么，问题可能是因为汇点-源点的接线问题，等等。记住检查信号的地线连接。还要核查脉冲的接通持续时间、脉冲保持时间与间歇时间之比、电平和频率是否符合输入技术要求。

2．**配置**—确保所有的配置参数都是正确的。**R7633** 必须被置为 **20**，**R7634** 必须被置为“**0002**”，使 **A** 相输入有效，**R7635** 必须被设为“**0000**”，使 **B** 相输入有效。

3．**保持在复位状态**—检查复位输入端 **I2** 的输入状态。如果 **I2** 为 **ON**，因为计数器被保持在复位状态，所以无法计数。

4．**梯形图程序**—保证在你的程序中使用计数器 **C76**。上面的输入端是计数器的计数允许端。在计数器计数前，它必须为 **ON**。中间的输入端是空端子，计数器要计数它就必须为 **OFF**。最后一个输入端是计数器复位，在计数时它必须为 **OFF**。

- **故障现象：计数器计数方向错误（加法计数变成减法计数）。**

可能的原因：

1．**通道 A 和 B 的分配**—编码器通道 **A** 和 **B** 的接线分配与要求的旋转/计数方向相反。只需交换输入端 **I0** 和 **I1**，计数方向就会反过来了。

- **故障现象：计数器可以加、减法计数，但不能复位。**

可能的原因：

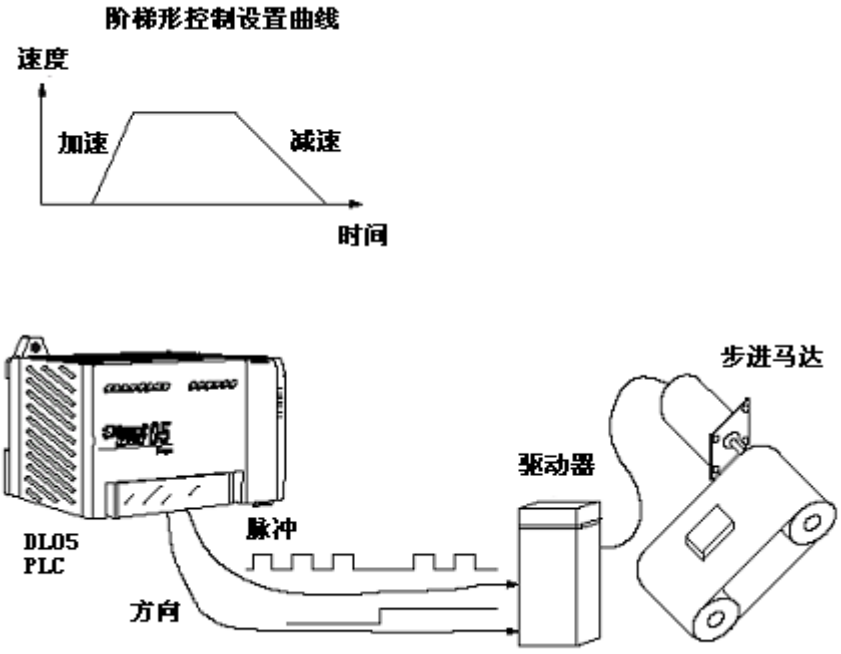
1．当你想要复位时，检查 **I2** 的状态指示灯，肯定它是工作的。还应核对检查 **I2** 的配置寄存器 **R7636** 已被置为 **7**。或者，如果你正在使用一个内部复位端的话，用 ***DireC*SOFT** 的状态方式监控计数器的复位输入端。

第四节　模式 30：脉冲输出

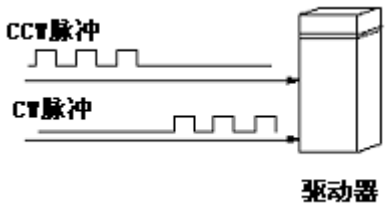
模式 30 中的 **HSIO** 电路会产生输出脉冲序列，它适用于单轴运动定位系统的开环控制。它发出脉冲和方向信号，这些可以连接到马达驱动系统上，来执行各种运动控制。使用模式 **30** 脉冲输出功能，你可以选择下面 3 种运动控制方式中的一种来实现你的控制系统。

- **阶梯形**—加速度梯度-定位速度-减速度梯度
- **中断控制**—中断的定位控制（也用于原点搜索移动）。
- **速度控制**—仅用于速度和方向。

在 **Mode 30** 下，**HSIO** 变成了一个脉冲发生器（最大达 **7KHz**）。用编程方法设定加速值和减速值、定位和速度目标值，**HSIO** 功能自动计算整个运动控制。下图说明了 **DL05** 可发出脉冲和定位信号来给步进定位系统的驱动器。脉冲输出功能独立完成所需控制要求，而不中断 PLC 中的梯形图程序的执行。

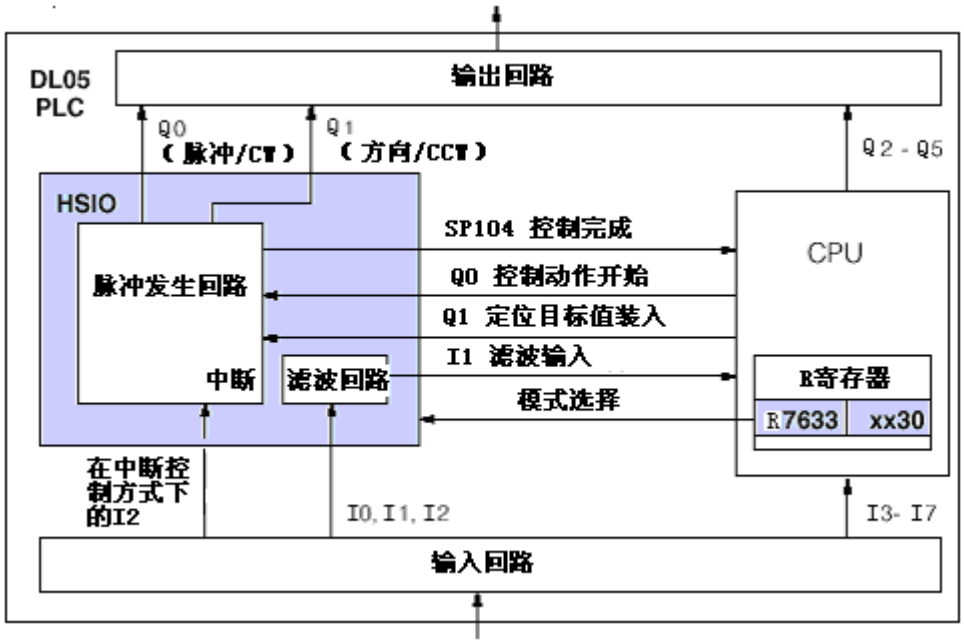


在上图中，**DL05** 产生脉冲和方向信号。每个脉冲代表定位系统运动的最小增量（例如步进系统的一步或一个微步）。或者，**HSIO** 脉冲输出方式可以被设置成反转（**CCW**）和正转（**CW**）脉冲信号，如下图所示。



注意：脉冲输出被设计用于开环步进电机系统。这一点，加上它的最小速度是 **40pps**，使它不适用于伺服电机的控制。

HSIO 在 **Mode 30** 时的逻辑组成图如下。



注意：必须是 DC 输出型的 DL05 才能使用脉冲输出功能。

在脉冲输出方式下，定义号 **Q0** 和 **Q1** 被重新定义或者说有两种不同的使用方式。物理的定义号是指螺丝接线端子，而逻辑定义号是指梯形程序图中的 **I/O** 定义号。请阅读下面几条，理解这一关键点。

- 在脉冲输出方式下，**I0** 和 **I1** 只能是滤波输入端，它们可作为梯形程序的输入触点。
- 在中断控制方式下，**I2** 用作时脉冲发生器的一个外部中断。在其它控制方式下，它可被用作滤波输入，就同 **I1** 一样（中断方式的配置如上所示）。
- 定义号“ **Q0** ”和“ **Q1** ”被用于两种方式。在物理输出连接端，**Q0** 和 **Q1** 端子把脉冲发送给运动系统。而梯形图程序使用逻辑定义号 **Q0** 和 **Q1** 启动模式 **30** 的“ **启动控制** ”和“ **装入位置值** ”的 **HSIO** 功能。

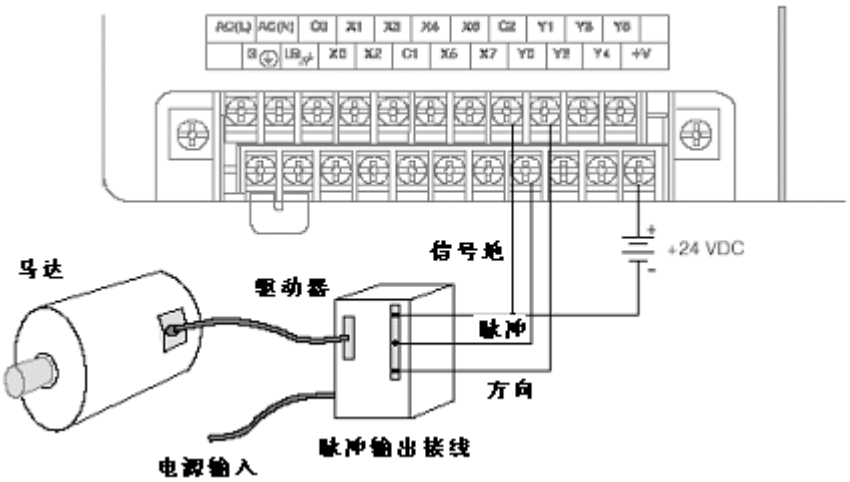
希望上面的讨论能解释为什么在脉冲输出方式下，一些 **I/O** 的定义名称具有双重含义。请仔细阅读这一段其余的部分，避免混淆所讨论的 **I/O** 的实际功能。

模式 30 的设定

使用设定模式 10 同样的方法，设置 R7633=0030，从而选择 HSIO 的模式 30 工作方式。

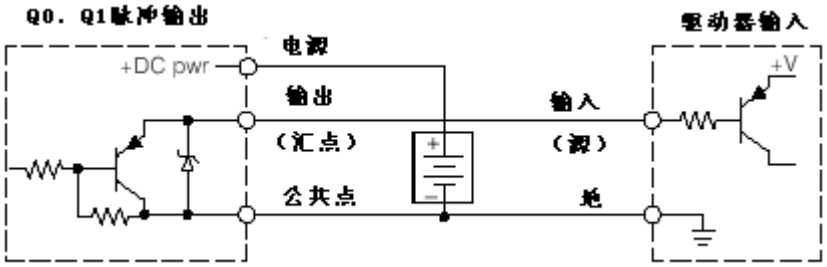
接线图

下面的通用接线图说明了脉冲输出端 **Q0** 和 **Q1** 连接到一个运动控制系统的驱动器输入端时的情况。

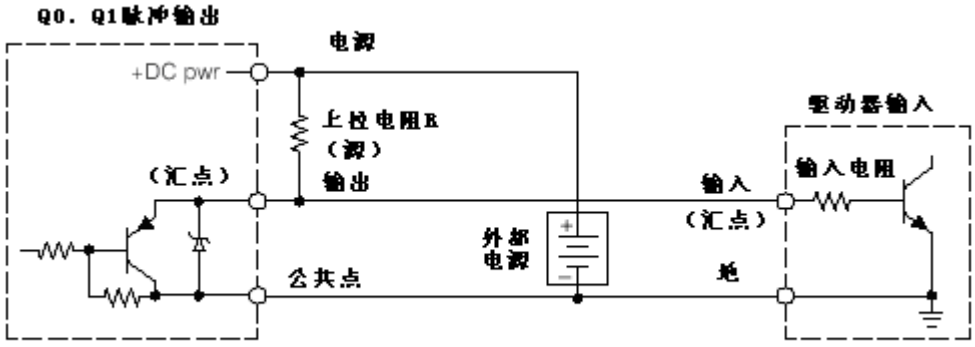


与驱动器输入端的连接

来自 PLC 输出端的脉冲信号 **Q0** 和 **Q1** 将如上所示连接到驱动器上。**注意，DL05** 的 **DC** 输出端仅为汇点型。下图说明了如何与一个源驱动输入回路连接。



下图显示了如何用一个上拉电阻与汇点驱动输入端连接。请参阅第 **2** 章，学会怎样计算和安装**上拉电阻**。



运转控制的技术参数

脉冲输出方式下产生的运转控制具有下列技术规格：

参数	技术规格
控制方式	阶梯形—加速度梯度/定位速度/减速度梯度
	中断控制—中断定位控制
	速度控制—仅速度和方向
位置范围	**-88388608-88388607**
定位方式	绝对/相对命令
速度范围	**40Hz-7KHz**
R 寄存器	**R2320-R2325**（控制参数表）
当前位置	**C76** 和 **C77**（**R1076** 和 **R1077**）

物理 I/O 配置

下表列出了脉冲输出方式下的 **I/O** 配置选择。**PLC** 使用 **SP104** 触点来识别 " **运动控制结束**"。**R7637** 用于选择脉冲输出的脉冲/方向或正转/反转方式。输入端 **I2** 专门用作中断控制方式中的外部中断。

物理 I/O	配置寄存器	功能	需要的十六进制码
-	**R7637**	**Q0**=脉冲 **Q1**=方向	**0103**
		Q0=正转脉冲 **Q1**=反转脉冲	**0003**
I0	**R7634**	软件滤波输入	**##06，**##=滤波时间 **0-99ms(BCD)**
I1	**R7635**	软件滤波输入	
I2	**R7636**	软件滤波输入	

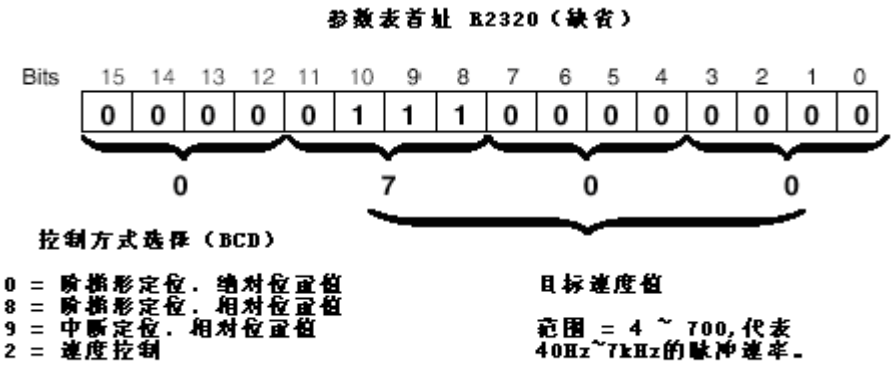
逻辑 I/O 功能

下面的逻辑 **I/O** 定义号定义了 **HSIO** 与梯形图程序之间的通讯功能。

逻辑 I/O	功能
SP104	**运动控制结束－**当运动控制完成时，**HSIO** 接通 **SP104**。当**启动控制**(**Q0**)接通时，再变成 **OFF**。
Q0	**启动控制－**梯形图程序接通 **Q0** 来启动控制过程。如果在控制任务完成之前，使 Q0 变为 OFF，则停止定位控制。再次置 Q0 为 ON，将启动另一个控制过程，除非当前位置等于目标位置。
Q1	**预设定位置值**—-如果处于运动停止状态，并且**启动控制** Q0 也为 OFF，此时你可把一个新值存入 **C76/C77** 中，然后接通 **Q1**。在转换时，**C76/C77** 中的值就变成当前位置。

定位/速度选择寄存器

控制过程参数表的第一个地址存贮了两个关键的信息。高四位（**12-15**）选择需要的控制类型。低 **12** 位（**0-11**）选择目标速度。



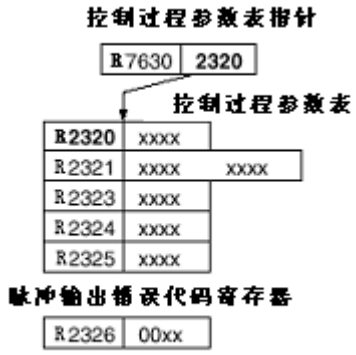
在建立三种控制方式的任意一种之前，梯形图程序必须设置这个地址的内容。**LDW** 和 **OUTW** 指令将写入全部 **16** 位，所以应保证每次完整地给出定位/速度选择寄存器的 4 位 **BCD** 值。

绝对位置值和相对位置值的选择确定了 **HSIO** 电路将如何解释你规定的目标位置值。绝对位置目标值是对应于位置 0 的目标位置值。相对位置目标值是对应于当前位置（前一目标位置）的目标值。你可以选择最适合的定义方法。

控制过程参数表

R7630 是一个指针，它指向控制过程参数表的开头。参数表的缺省起始地址是 **R2320**。但是，你可以用编程的方法在 **R7630** 中设定不同的值来改变它。记住用 **LDR**（读出地址）指令，把八进制地址变成十六进制值。

各定位控制方式下，**HSIO** 的参数表的定义如下。其最后一个寄存器（缺省为 R2326）为错误代码寄存器，当你启动某个定位控制过程时，如果参数表的设置有错，则会在其错误代码寄存器中写入错误代码。错误码的定义祥见本节末尾的错误代码表。



阶梯形控制方式下的参数表（缺省情况下）

R 寄存器	功能	范围	单位
R2320，12-15 位	阶梯形控制	**0**=绝对值方式， **8**=相对值	—
R2320，0-11 位	目标速度值	**4-700**	× **10pps**
R2321/R2322	目标位置值	**-8388608-8388607**	脉冲数
R2323	启动速度	**4-100**	× **10pps**
R2324	加速时间	**1-100**	× **100 mS**
R2325	减速时间	**1-100**	× **100mS**
R2326	错误代码	（见本节末尾）	—

中断控制方式下的参数表（缺省情况下）

R 寄存器	功能	范围	单位
R2320，12-15 位	中断控制	**9**=相对值	**—**
R2320，0-11 位	目标速度值	**4-700**	**× 10pps**
R2321/R2322	目标位置值	**-8388608-8388607**	脉冲数
R2323	启动速度	**4-100**	**× 10pps**
R2324	加速时间	**1-100**	**× 100 mS**
R2325	减速时间	**1-100**	**× 100mS**
R2326	错误码	（见本段末尾）	**—**

速度控制方式下的参数表（缺省情况下）

R 寄存器	功能	范围	单位
R2320	速度控制	仅 **2000**	**—**
R2321/R2322	方向选择	**80000000**=CCW **0**=CW	脉冲数
R2323	起动速度	**4-700**	**× 10pps**
R2326	错误码	（见本节末尾）	**—**

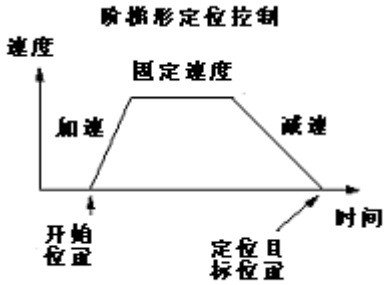
选择控制的类型

脉冲输出方式产生三种类型的运动过程控制。大多数应用情况下，所有的运动采用同一种控制方式。但是，如果要求的话，每一个运动过程多可采用不同的过程控制方式

- **阶梯形**—加速度梯度-定位速度-减速度梯度
- **中断控制**—中断的定位控制（以规定的速度运动到目标位置）。
- **速度控制**—仅用于速度和方向。

阶梯形过程控制

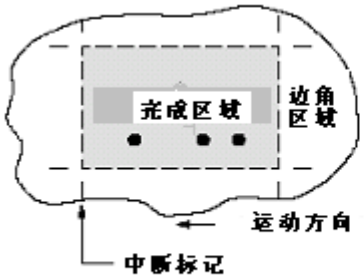
阶梯形控制是最普通的定位控制。通过建立一个定位控制过程，它把被定位物体移动到一个预定的目标位置。在定位控制开始位置使用加速度梯度控制。使用减速度梯度控制的开始位置由定位目标位置等推算得出。中间的位移部分以恒定的速度行进。当移动的起始位置和末端位置之间的距离已知时，阶梯形控制最适用于简单的点-点移动控制。

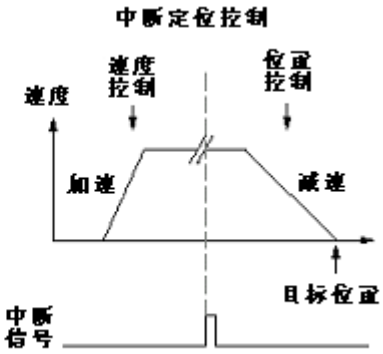


中断定位控制和原点搜索控制

在一些应用中，产品的加工材料要移到加工刀具旁，例如钻孔台。如下所示，在被加工件边角区域的标记可让设备刀具按矩形记录下它的位置，然后正确钻孔。原点搜索定位可让开环运动系统在上电时重新测量（重新设定）当前位置值。

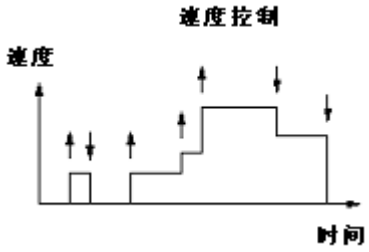
中断控制结合了速度控制方式和位置控制方式。定位开始后，通过预定的加速度达到一个设定的速度。速度是持续的，移动是不连续的。当接受到外部中断信号后（由于识别到了中断），控制就从速度控制转到了位置控制。它连续运行一段预定的距离至目标点（例如一个钻孔位置），结束定位运动。在到达目标位置前使用减速度梯度控制。





速度控制

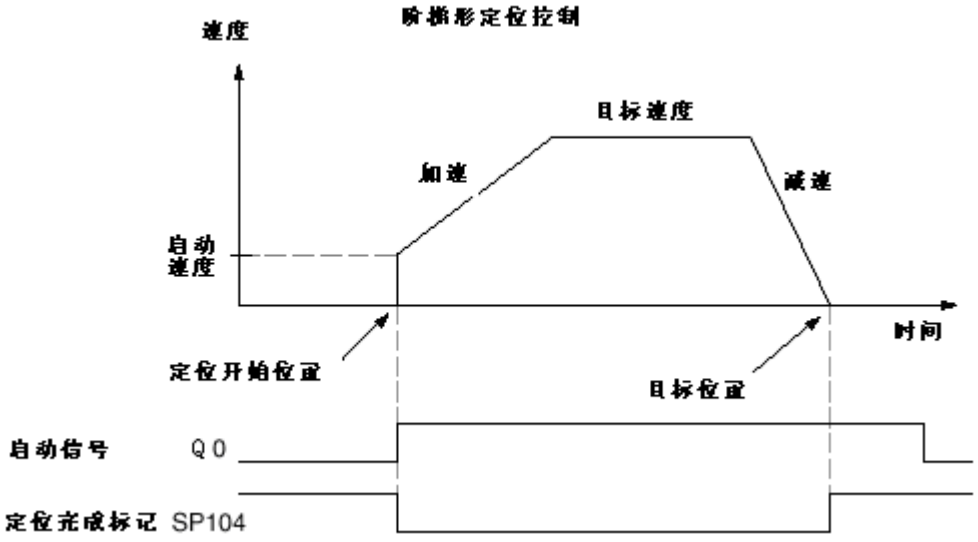
速度控制方式仅控制定位的方向和速度。在该种方式下没有位置目标值，所以定位距离不定。只有第一个运动速度值需要预设定，其余的速度值可以在运动过程中被创建。右图箭头所示指出了速度的变化。



阶梯形定位控制

阶梯形定位控制的应用

阶梯形定位控制最适用于当目标位置已知时简单的点到点运动。定位起始速度的范围必须在 **40pps-1kpps** 之间。其它的控制参数在控制参数表中设定。



在控制曲线下的时序图表示了定位事件的次序。

HSIO 使用逻辑输出 **Q0** 作为阶梯型定位控制的启动输入。Q0 信号一成立，**HSIO** 立即使定位完成标记 SP104 失效，所以梯形图程序可以监控定位控制的进程。通常梯形图将监控该标记位，以便知道什么时候开始第二次定位控制运动。

如果你熟悉了定位控制，你会注意到我们不必规定运动的方向。HSIO 功能会检查对应于当前位置的目标位置，并且自动地把正确的方向信息输出给电机驱动器。

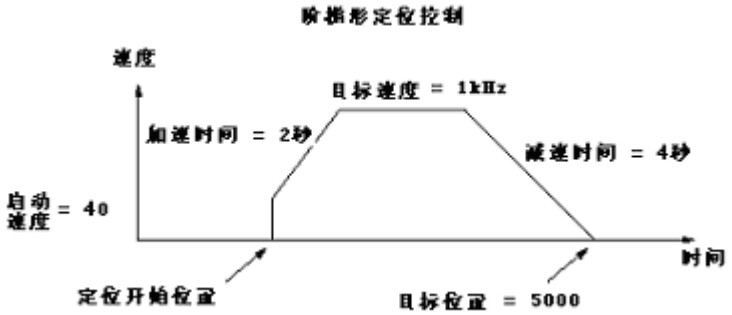
注意定位运动会立即加速到启动速度，该点在步进系统中非常有用。在由于低力矩问题或电机中的共振点引起熄火时，利用该特点，我们就可以跳过低速区。（当步进电机熄火时，我

们就丢失了开环定位系统中的负载位置）。但是最好不要使启动速度太大，因为由于系统的惯性，步进电机将会“ 漏掉 ”一些脉冲。

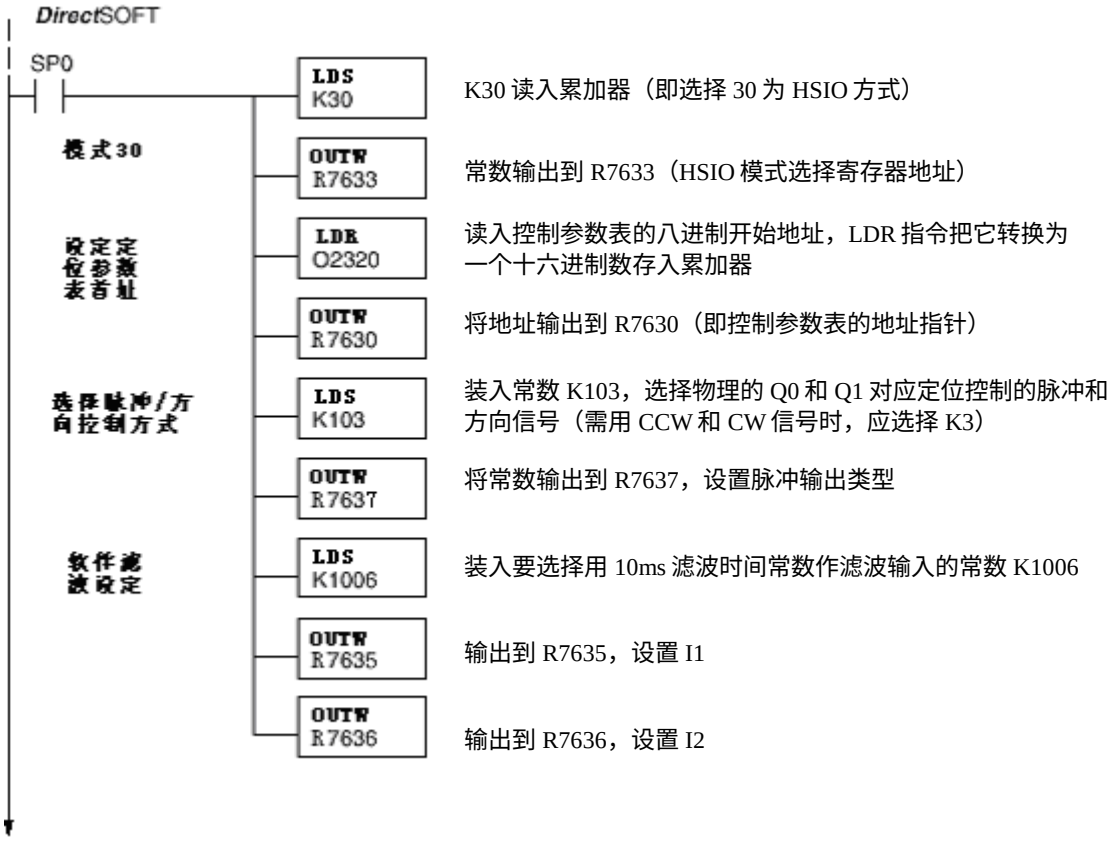
当你需要改变当前位置值时，用逻辑输出线圈 **Q1** 把一个新的值存到 **HSIO** 计数器中。如果梯形图程序把一个新值存到了 **C76/C77**（**R1076/R1077**）中，然后置位 **Q1**，则系统将把该值传送到 **HSIO** 电路的计数器中。这些工作必须在控制开始之前进行，因为在定位运动中，**HSIO** 将忽略 **Q1** 信号。

阶梯形定位控制的程序例子

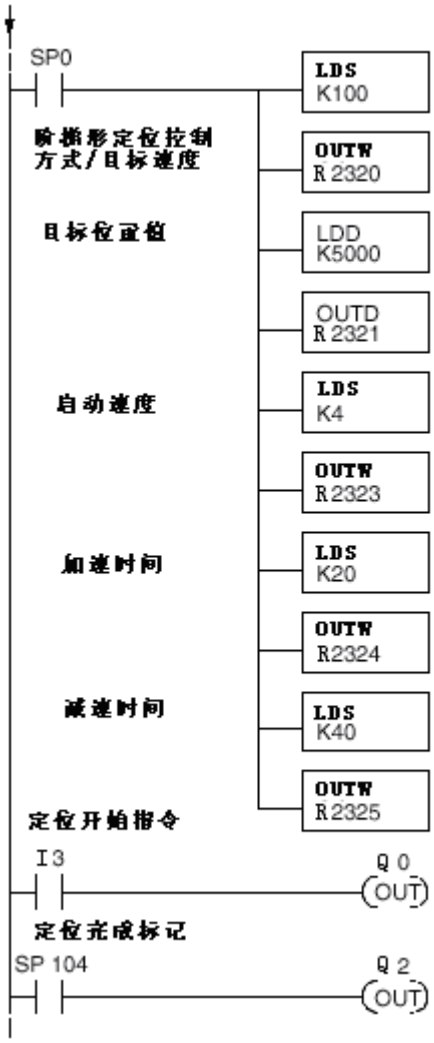
下图画出并标明了我们想要执行的阶梯形控制曲线。它包括一个非零启动速度和适度的目标速度。



执行下面的程序将能实现上面画出的控制曲线。程序的开头部分是对模式 30 脉冲输出方式的必要的参数设置。因为这在程序中只需运行一次，所以我们使用初始接通扫描触点 **SP0** 来触发设置工作。



（接上页）



读入要选择绝对位置型阶梯形控制、和 1kHZ 目标速度的常数 K100

常数输出到 R2320（控制选择/启动速度设置寄存器地址）

装入 5000 脉冲的目标位置值，为双字节（8 位 BCD 位置值）

将此值输出到 R2321 和 R2322 中（即目标位置寄存器地址）

读入要选择 40HZ(4×10pps)的启动速度的常数 K4

输出到 R2323，即启动速度参数寄存器地址

读入要选择 2 秒(20×100ms)加速时间的常数 K20

输出到 R2324，即加速时间参数寄存器地址

读入要选择 4 秒(40×100ms)减速时间的常数 K40

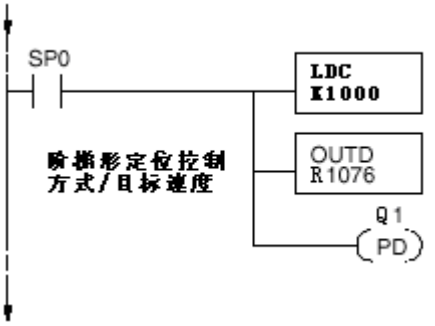
输出到 R2325，即减速时间参数寄存器地址

接通 I3，逻辑输出 Q0 启动控制过程

使用 Q2 启动一个蜂鸣器响以表示定位控制过程完成

设定位置值

你可以在任何时候把一个新的位置值设定成当前位置值。通常在完成原点搜索之后进行该工作（参见中断程序举例）。



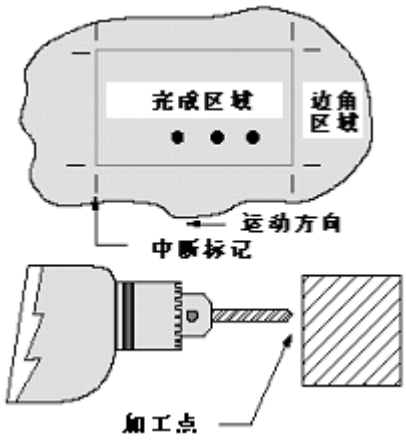
读入常数 K1000 作为新的当前位置值

输出到 R1076/R1077(C76/C77)，即当前位置值地址

接通 Q1 一个扫描周期，在它从 OFF 变为 ON 的上升沿，HSIO 用该值预设为当前位置值

中断定位控制

中断定位的应用

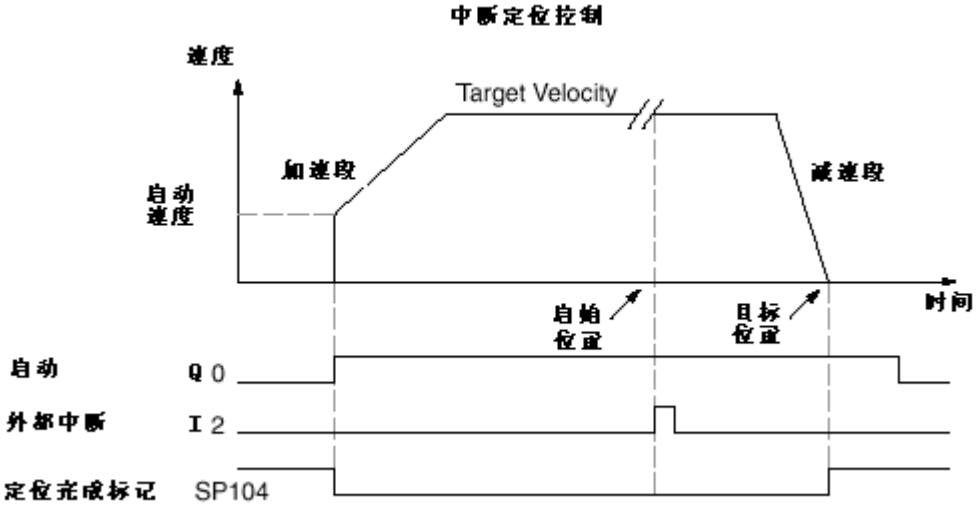


1． 在如右所示的典型的应用情况下，产品的加工材料移到加工刀具旁，例如是一个钻头。在工件边角区域的标记处让设备刀具按矩形记录下它的位置，然后正确钻孔。

2． 在其它定位例子中，工件是固定不动的，刀具是移动的。钻头靠近加工工件的表面，准备钻一个精确深度的孔。但是，钻头的长度会由于刀具的磨损而逐渐减小。克服这一问题的方法是，每次钻孔时，检测接触到加工件表面的时刻，在碰到后再往工件钻入一段固定的距离。

3． 原点搜索定位可以使运动系统测量到它在启动时的位置。在这种情况下，定位系统先进行一段不确定的移动，一直等到它碰到了原点限位开关。在此确定位置会产生一个中断。然后我们停止定位运动，用一个等于实际“ **原点位置** ”的数字来设置好位置值。

定位控制只能从速度控制开始。一旦物理输入端 **I2** 上产生了中断脉冲信号，定位启始位置就被当作当前位置。速度控制变成位置控制，把负载从当前位置定位到目标位置。注意最小启动速度为 **40pps**。这个瞬时速度提供给在低速时可能会熄火的步进电机。

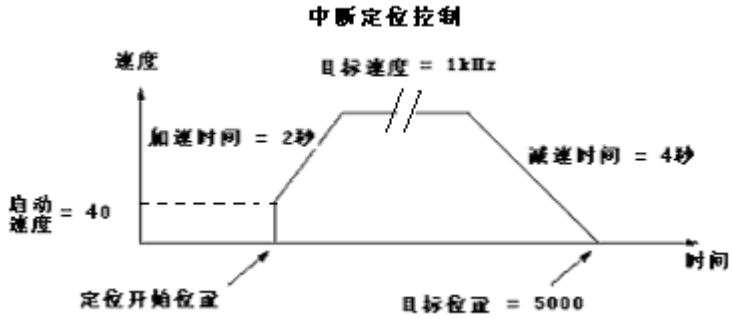


上图中，在控制曲线下的时序图说明了定位事件的次序。

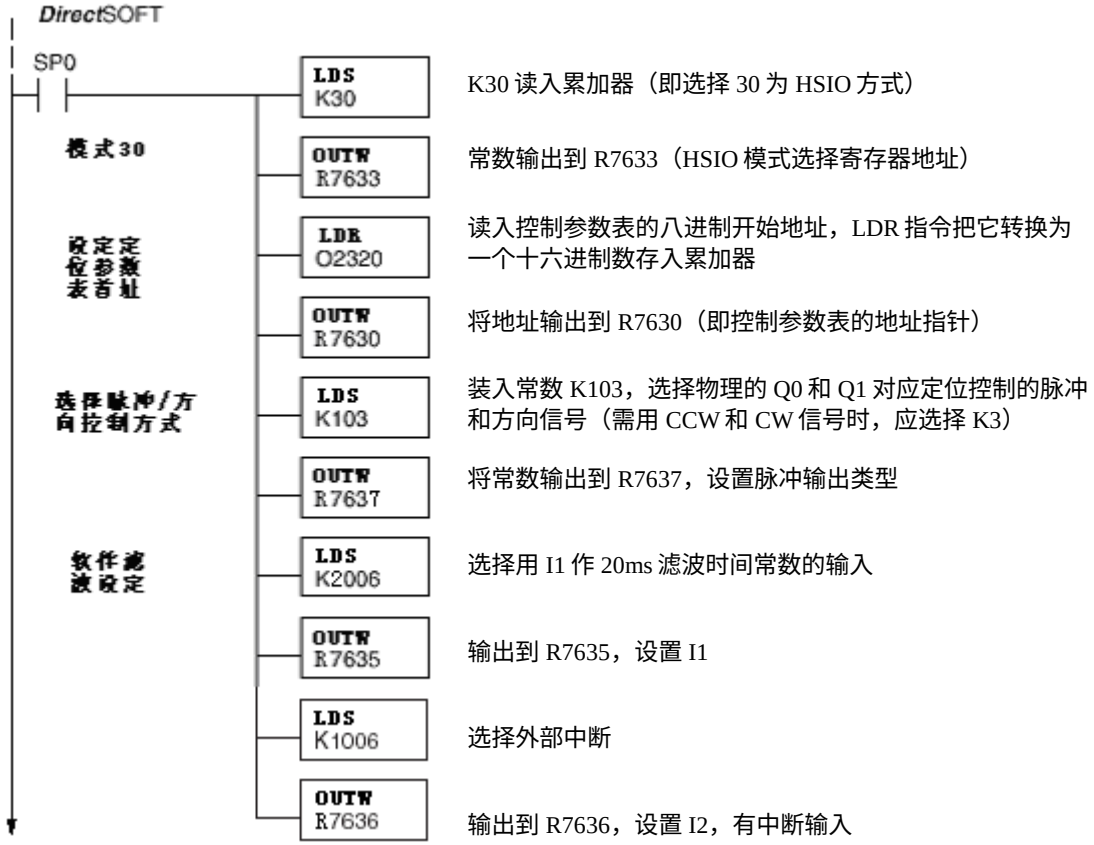
CPU 使用逻辑输出端 Q0 启动控制。一旦定位运动启动，**HSIO** 立即复位定位完成标记（**SP104**），所以，通过识别该信号的状态，梯形图程序可以监控定位控制的完成情况。

中断定位控制程序例子

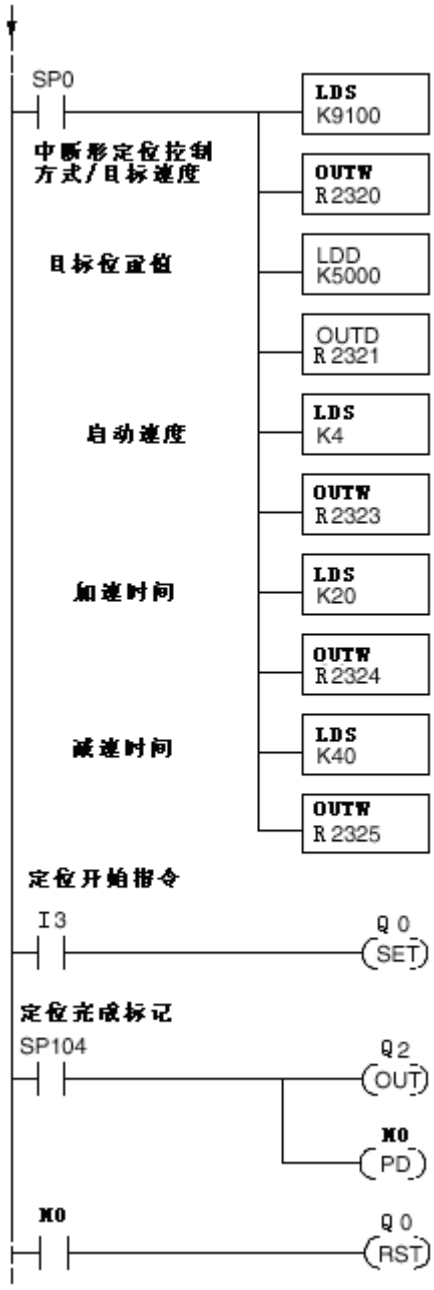
下图画出并标明了我们想要执行的中断定位控制曲线。它包括一个非零起始速度和适中的目标速度。



执行下面的程序将能实现上面画出的控制曲线。程序的开头部分是对模式 30 脉冲输出方式的必要的参数设置。因为这在程序中只需运行一次，所以我们使用初始接通扫描触点 **SP0** 来触发设置工作。



（接上页）



读入要选择中断形控制、绝对位置和 1kHZ 的目标速度的常数 K9100

输出到 R2320，即控制选择/启动速度设置寄存器地址

选择 5000 脉冲的目标位置值，要使用双字节（8 位 BCD）

将常数输出到 R2321 和 R2322(目标位置值双字节寄存地址)

选择 40HZ(4×10pps)的启动速度

输出到 R2323，即启动速度参数寄存器地址

选择 2 秒(20×100ms)加速时间

输出到 R2324，即加速度参数寄存器地址

选择 4 秒(40×100ms)减速时间

输出到 R2325，即减速度参数寄存器地址

I3 是一个点动开关，当 I3 为 ON 时，Q0 启动控制

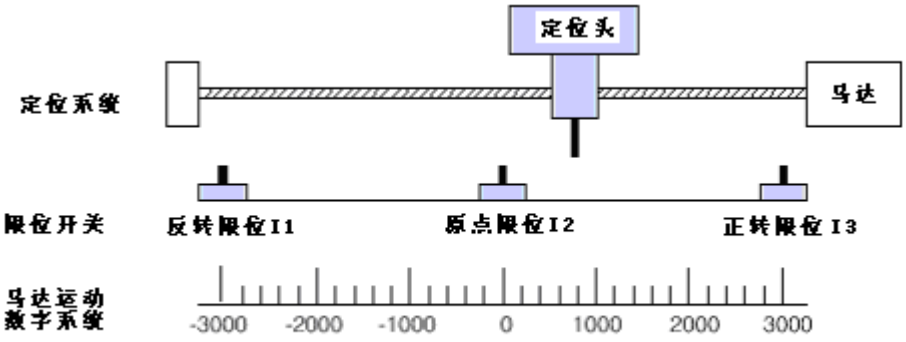
SP104 是 HSIO 指示定位完成的标记线圈，我们使用 Q2 启动一个蜂鸣器响以表示控制完成

M0 接通一个扫描周期，复位 Q0，停止定位控制。

置位启动输入 **I3**，定位过程动作就开始了。然后开始移动一段不定距离，直到 **I2** 上出现了外部中断信号。然后继续移动 **5000** 多个脉冲距离，完成定位过程。

原点搜索程序例子

运动控制较具挑战性的一面是上电时实际位置的建立。这点对于没有位置反馈装置的开环系统尤为如此。然而，在定位机构的某个确定位置安装一个简单的限位开关，就可以提供单点“**位置反馈**”。对于大多数步进控制系统，这是一个好的、经济的解决办法。



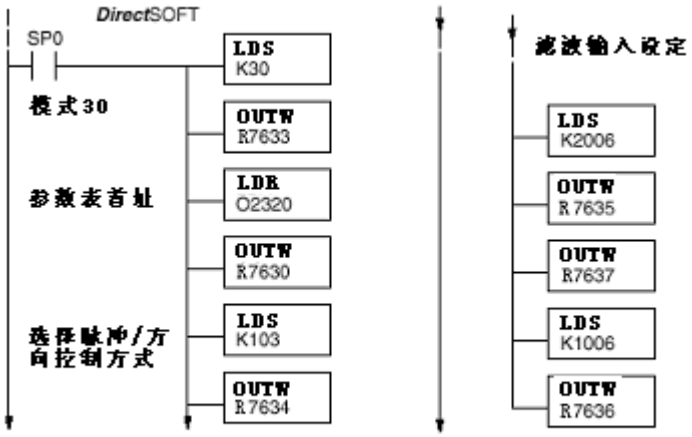
在上图中，根据电机的反转/正转，定位头左/右移动。在定位头的移动超过界限并将损坏设备前，**PLC** 梯形图程序会识别到反转/正转的限位开关，使电机停转。原点限位开关用于上电时建立实际位置值。

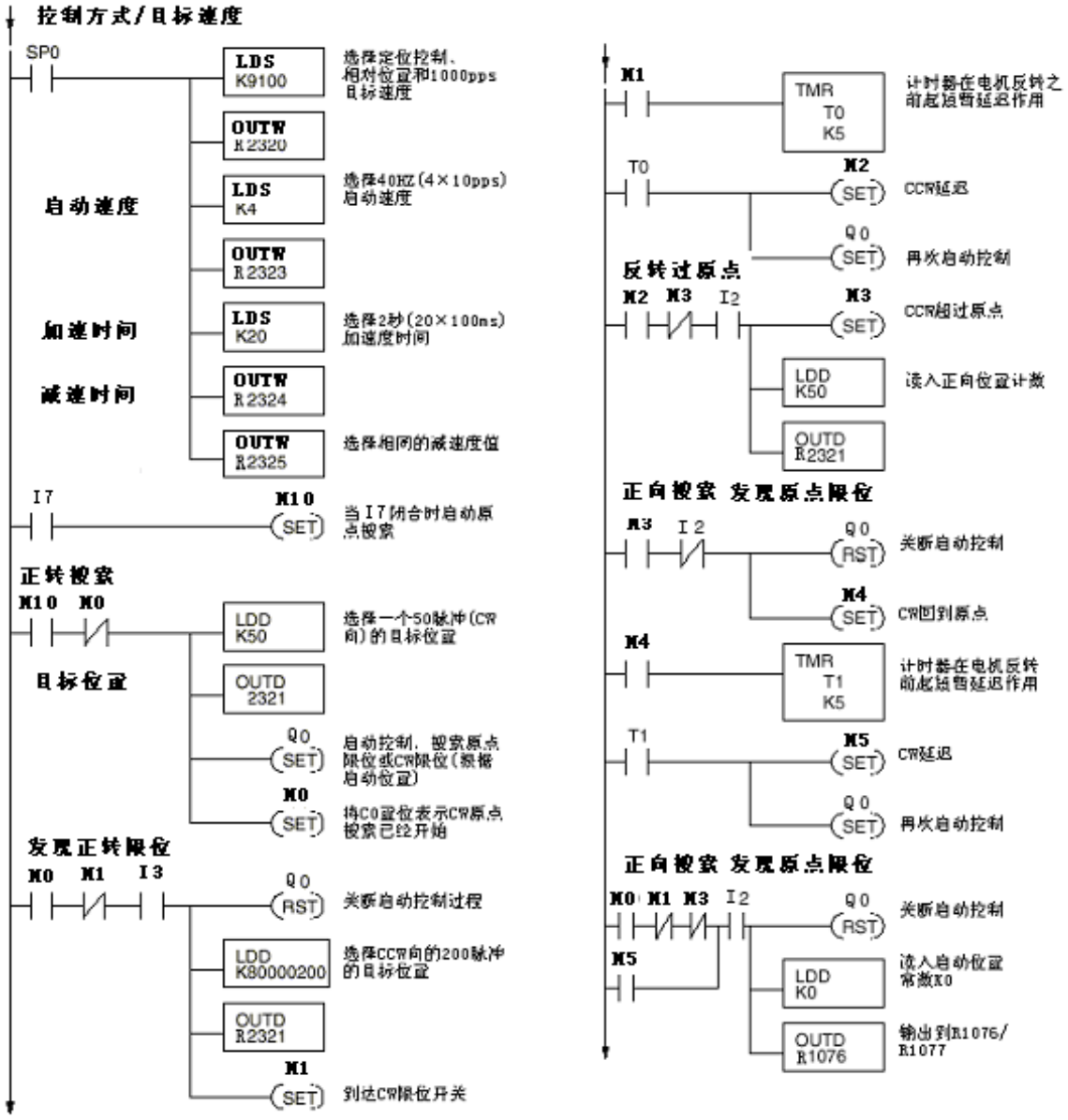
上电时，我们不知道定位头在原点限位开关的左边还是右边。因此，我们要用中断方式建立一个原点搜索控制。原点限位开关被接到 **I2** 上，它将引发中断。我们选择任意方向开始搜索，如按正转方向（从左到右）移动。

· 如果第一次就碰到了原点限位，那么就可以停下来，把位置初始化（此值一般为“**0**“，如果你喜欢的话，也可以是其它不同值）。

· 但是，如果第一次碰到了正转限位开关，则我们必须让电机反转，直至碰到原点限位开关，再稍稍移动一点就停下来。

在后一种情况下，我们还要重复第一次的移动，因为我们总是需要***从同样方向***最后靠近原点限位开关，所以，最后的实际位置在任何情况下都是相同的。



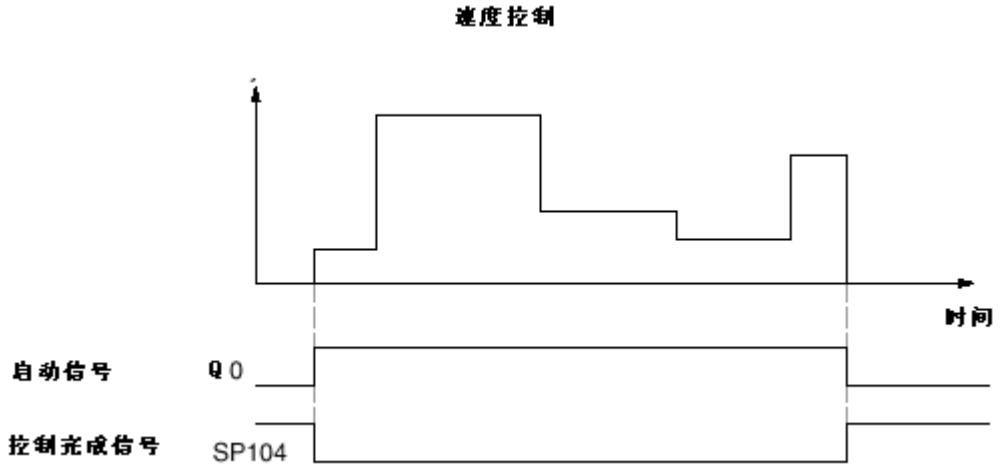


根据检测到限位开关的次序，原点搜索控制将执行程序的不同部分。梯形图程序置位 M**0** 开始正转方向的原点搜索。如果遇到了正转限位，程序则反转方向搜索原点，一过原点，又以正转方向完成最后的原点搜索。到达原点后，最后的梯形图回路把当前位置值设成“ **0** ”。

速度控制

速度控制的应用

速度控制最适用于不需要移到规定点的运动控制应用。传送带速度控制是一个典型的应用例子。



上图中，在控制曲线下的时序图说明了控制事件的次序。

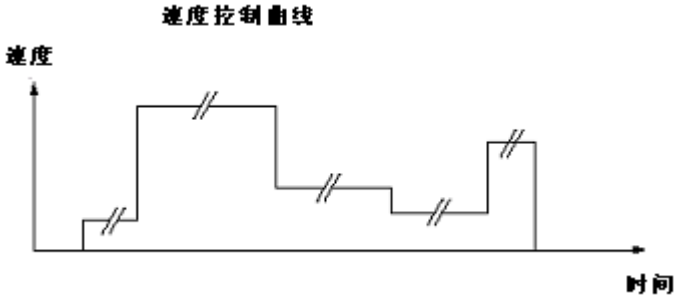
假设设定的速度大于零，启动信号（**Q0**）成立则开始速度控制运动。因为没有最后的定位目标位置，所以只要启动信号有效，就一直处于速度控制中。

在启动信号有效期间，通过把一个新的值写到速度寄存器（缺省为 **R2323**），程序可以随时改变控制速度。有效的速度范围是 **40Hz-7kHz**。从图中可注意到，速度的变化之间没有加速度或减速度的梯度变化。这就是用 **HSIO** 进行速度控制的结果。但是，通过更缓慢地增加或减少速度值，程序可以控制更多级的速度变化。计数器或定时器可以用来建立你自己的加速/减速梯度。除非定位头必须进行一种极复杂的移动，否则的话，选择阶梯形定位控制或中断定位控制，让 **HSIO** 功能来产生加速/减速梯度将更容易一些。

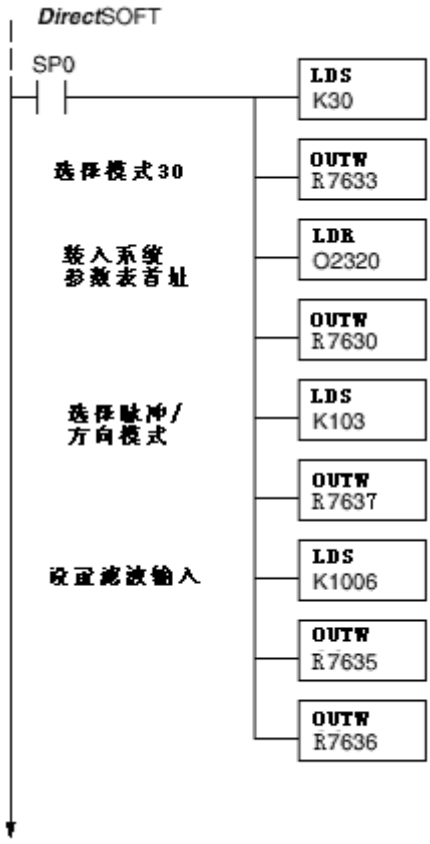
与阶梯形定位控制和中断定位控制不同的是，你必须规定速度控制时行程要求的方向。反向时，把 **8000 0000 （HEX）**存入方向选择寄存器（缺省 **R2321/R2322**），或正向时送 **0**。

速度控制的程序例子

下图画出并标明了我们想要执行的中断控制曲线图。每一速段都是不定长度的。当使用逻辑程序（或其它手段写到 **R** 寄存器）改变了速度参数，控制速度才变化。



下列程序中使用了各自独立的输入端来装入各新的速度值。这是一段有趣的程序，因为只用两三个输入开关，你就可以建立一种无限速度控制。所要注意的是一次只能接通 I2，I3 或 I4 中的一个。程序的开头包括所有脉冲方式下的必要的设置参数。在程序中这项工作只需做一次，所以我们使用初始扫描接通触点 **SP0** 来触发设置工作。



K30 读入累加器，即选择模式 30 为 HSIO 方式

输出到 R7633，即 HSIO 方式选择寄存器地址

读入控制参数表的八进制起始地址

将此地址输出到 R7630（控制参数表地址指针）

选择 Q0 和 Q1 为脉冲和方向功能

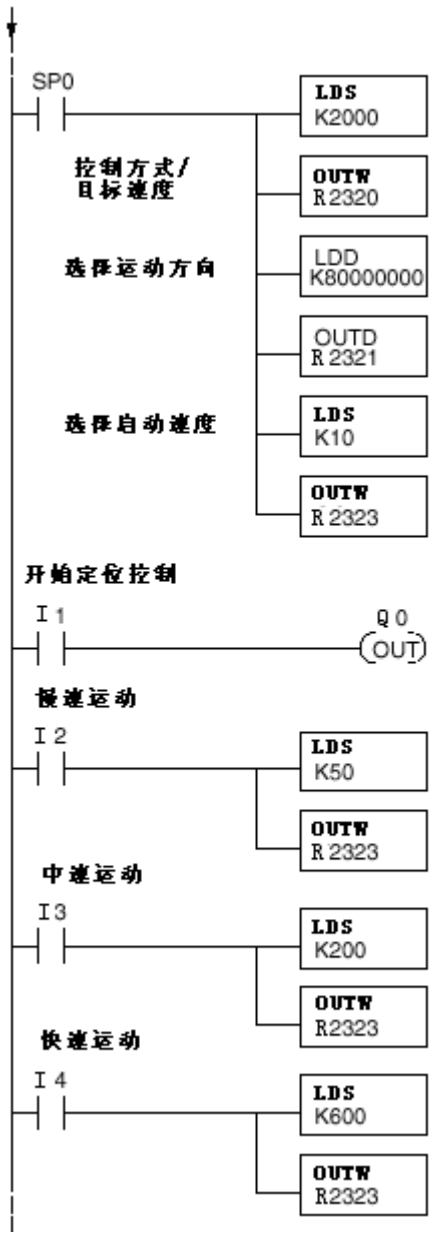
将常数输出到 R7637，设置脉冲输出类型

选择 10ms 滤波时间常数的滤波输入

将常数输出到 R7635，设置 I1 滤波时间

将常数输出到 R7636，设置 I2 滤波时间

（接上页）



读入要选择速度控制的常数 K2000

常数输出到 R2320（即控制参数表）

读入选择 CCW 方向的速度控制常数 K80000000，别忘记使用双字节（8 位 BCD 位置值）

将此常数输出到 R2321 和 R2322（即目标位置双字节寄存器地址）

读入要选择的初始速度 100pps 常数 K10

输出到 R2323，即速度参数寄存器地址

当 I1 定位条件成立，逻辑输出 Q0 启动控制过程

I2 闭合，读入 500pps 速度的常数 K50

输出到 R2323，即速度参数寄存器地址，速度将立即改变

I3 闭合，读入 2000pps 速度的常数 K200

输出到 R2323，即速度参数寄存器地址，速度将立即改变

I4 闭合，读入 6000pps 速度的常数 K600

输出到 R2323，即速度参数寄存器地址，速度将立即改变

脉冲输出模式的错误代码

R2320（缺省地址）开头的控制参数表决定了运动控制方式。当 **HSIO** 试图根据参数表的设定执行定位控制时，某些非法的设置数字将导致错误。当发生错误时，**HSIO** 将错误码写入 **R2326** 中。

错误码	错误说明
0000	无错误
0010	所要求的控制类型码是无效的（必须使用 **0，1，2，8，**或 **9**）
0020	目标速度不是 **BCD** 码
0021	设定的目标速度低于 **40pps**
0022	设定的目标速度大于 **7000pps**
0030	目标位置值不是 **BCD** 码
0040	启动速度不是 **BCD** 码
0041	设定的启动速度低于 **40pps**
0042	设定的启动速度大于 **1000pps**
0050	加速时间不是 **BCD** 码
0051	加速时间为零
0052	加速时间大于 **10** 秒
0060	减速时间不是 **BCD** 码
0061	减速时间为零
0062	减速时间大于 **10** 秒

通过再次检查控制参数表的值，大多数错误可以被纠正。在上电时，以及在从编程方式切换到运行方式时，错误会自动被清除。

Mode 30 的故障检修指导

如果你在使用 HSIO 的模式 **30** 时遇到了问题，请仔细阅读下列故障现象和可能的原因。最常见的问题被列出如下：

故障现象：步进电机不转。

可能原因：

1．**配置**—检查 **HSIO** 在输出端 **Q0** 和 **Q1** 上实际产生的脉冲。当你启动定位控制时，监视 **Q0** 和 **Q1** 发光指示灯的状态。如果发光管闪烁或常亮，配置可能是正确的。

2．**编程错误**—如果 **Q0** 或 **Q1** 上没有脉冲，可能存在编程错误。检查当 **PLC** 进行定位控制时，是否在 **R2326** 中产生了错误码。错误码说明如上所示。

3．**检查目标值**—如果计数值等于目标值（**count=0, target=0**），就没有脉冲。

4．**接线**—检查与步进电机的接线是否正确。记住从 **PLC** 到运动系统间需要信号接地连接。

5．**运动系统**—检查确认驱动器已上电且有效工作。你可以用模式 **60**（普通输入/输出）编制简单的 I/O 通断程序来查明运动系统是否正常工作。例如，你可以用 **I0** 和 **I1** 手动控制对应的 **Q0** 和 **Q1**，从而控制电机任一方向的单步运行。如果用这种简单的控制方式，电机也不转的话，那么在使用模式 **30** 前，请首先解决电机驱动系统的问题或查看接线是否正确。

6．**存储器错误**—**HSIO** 配置参数被存贮在 **CPU** 系统存储器中。某些 **HSIO** 操作有时会破坏这些数据。如其他正常操作都无法解决问题的话，把系统存储器初始化，可能会解决问题。用 ***Direct* SOFT**，选择 ***PLC*** 菜单，然后选择 ***Setup***，然后 ***Initialize Scratchpad***。

故障现象：电机运转方向错误。

可能的原因：

1．**接线**—如果你选择了 **CW** 和 **CCW** 型操作，那么只要调换输出端子 **Q0** 和 **Q1** 的接线。

2．**方向控制**—如果你选择了**脉冲**和**方向**型操作，那么只要把方向位置反即可。

第五节　模式 40：外部中断

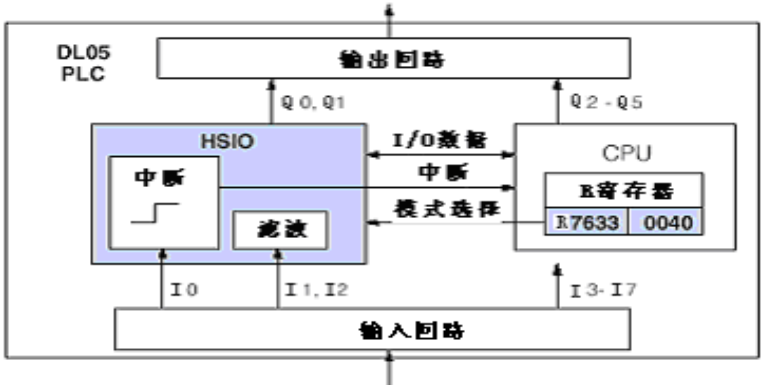
HSIO 模式 40 给程序提供了一个高速中断。该性能供你作下列应用选择：

· 一个外部事件需要触发 **CPU** 的一个中断子程序，在中断程序中一般用直接 I/O 指令，这称为外部中断。

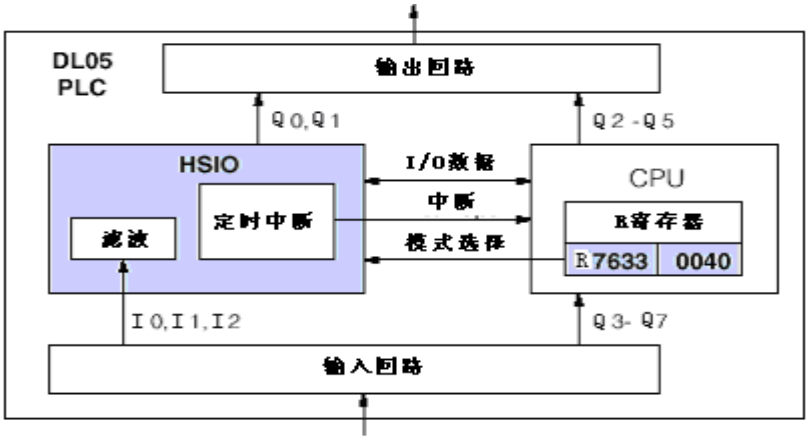
· 按不同于 **CPU** 扫描时间的时基（比它快或慢）执行中断程序。中断时间可用程序控制，从 **5-999mS** 可选，这称为定时中断。

逻辑功能图

HSIO 电路建立了对 **CPU** 的高速中断。下图给出了使用 **I0** 的外部中断选择。在此配置中，**I1** 和 **I2** 是普通滤波输入。



或者，你可以把 **HSIO** 电路设置成按定时器时间来产生中断，如下所示。在此设置中，输入端 **I0-I2** 是滤波输入。



模式 40 的设定

在 HSIO 模式选择寄存器 **R7633** 设置 BCD 码 40，选择**模式 40**。

R7633 = 0040 （BCD）

模式 40 下，I0~I2 的设定

下表列出了高速中断方式下 I0~I2 的输入选项设置。当 **R7634** 中为 “ **0004** ” 时，输入端 **I0** 为外部中断。而如果你需要用定时中断功能时，那么在 **R7634** 中设置定时中断时间，此时输入端 **I0** 变成了一个滤波输入点（它使用 I1 的滤波时间常数）。在高速中断方式下，输入端 **I1** 和 **I2** 只能是用作滤波输入，有各自的滤波时间设置寄存器，可分别设置滤波时间常数。记住，当选择了定时中断功能时，**I0** 的滤波时间常数使用 **I1** 的滤波时间常数。

输入	配置寄存器	功能	需要的十六进制码
I0	R7634	外部中断	**0004**（缺省）
	使用 **R7635** 中 I1 的时间设定值	滤波输入（当使用定时中断功能时）	×××**4,** ×××=定时中断时间 **5-999ms(BCD)**
I1	R7635	滤波输入	××**06(**××=滤波时间**)0-99ms(BCD)**
I2	R7636	滤波输入	××**06(**××=滤波时间**)0-99ms(BCD)**

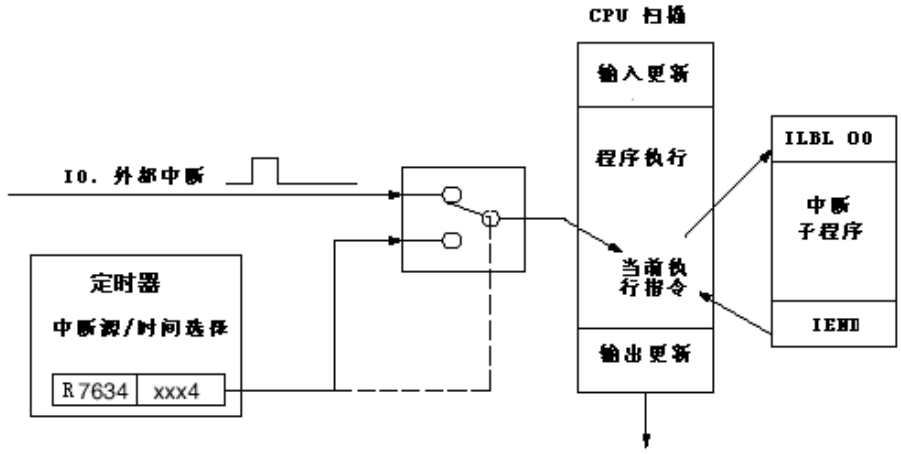
中断和梯形图程序

参阅下图。中断源可能是外部的（**I0**），或是 **HSIO** 定时器功能。**R7634** 中设定的参数具有双重含义：

· 它可选择两种中断源，外部（**I0**）或内部定时器。

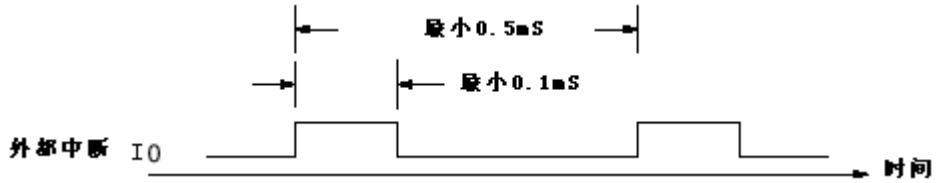
· 在定时中断情况下，定时中断的时间在 **5~999mS** 间可选。

中断子程序用 **ILBL O0** 定义。记住在你的程序开头应含有 **INE** 指令。否则，中断子程序将无法`执行。



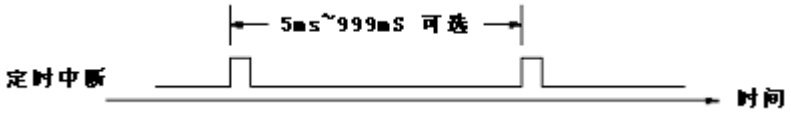
外部中断时序

I0 的信号脉冲必须符合某个定时标准，以保证中断的产生。参阅下面的时序图。**I0** 的输入特性是固定的（它不是一个程控的滤波输入）。最小脉冲宽度为 **0 .1mS**。在第二次中断脉冲到来之前，必须有一段延时，所以中断周期不可能小于 **0 .5mS**。



定时中断时序

当选择了定时中断时，**HSIO** 产生中断信号。此时中断没有“ 脉宽 ”，但是可在 **5-999mS** 之间调节中断周期。



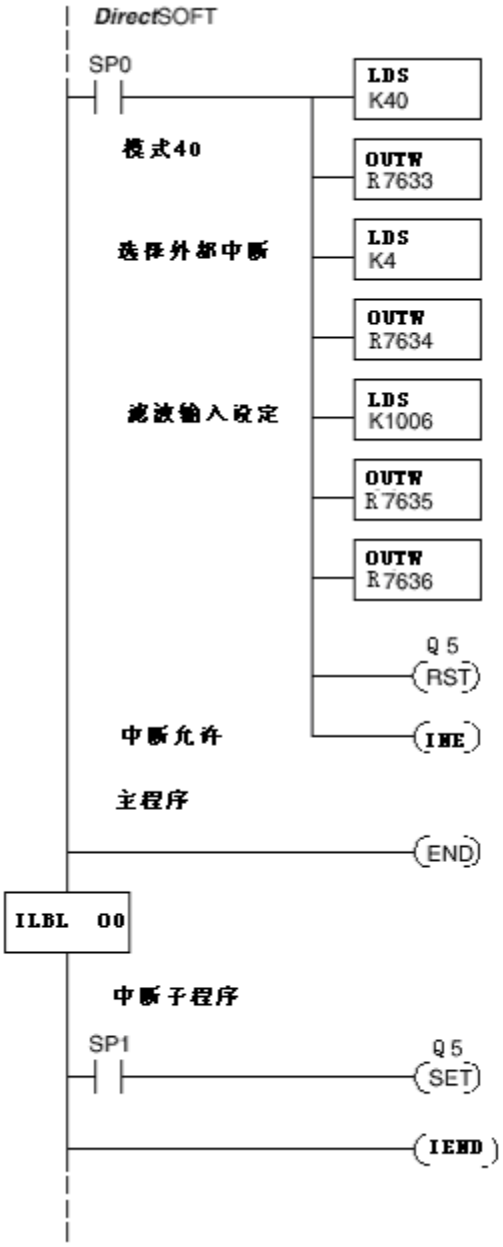
独立的定时中断

中断 **O1** 也被作为一个中断。这个中断与 **HSIO** 特性无关。中断 **O1** 使用一个被设置在 **R7647** 中的内部定时器来启动。中断周期 **5-9999m**S 调节。一旦中断周期被设置好了，并在程序中允许中断，**CPU** 将按照 **R7647** 中的时间设定值调用定时中断子程序。

输入	配置寄存器	功能	需要的十六进制码
-	R7647	高速定时中断	××××(××××=定时器设定值)**5-9999mS（BCD）**

外部中断程序例子

下面的程序选用了模式 40，然后选择了外部中断。输入端 I1 和 I2 被设置成滤波输入端，时间常数为 10mS。这段程序是通用的，可为你所用。



K40 读入累加器，即选择模式 40 为 HSIO 方式

输出到 R7633（HSIO 方式选择寄存器地址）

选择外部中断，I0 为中断输入

输出到 R7634，设置 I0 的外部中断操作

选择 10ms 滤波时间常数作滤波输入的常数

输出到 R7635，设置 I1

输出到 R7636，设置 I2

复位 Q5

中断允许

在这里插入要应用的主程序

主程序结束

中断服务程序开始（ILBL O0）

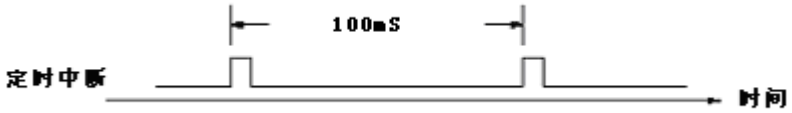
在这里插入中断服务程序

置位 Q5，模拟对应于 I0 的脉冲捕捉功能

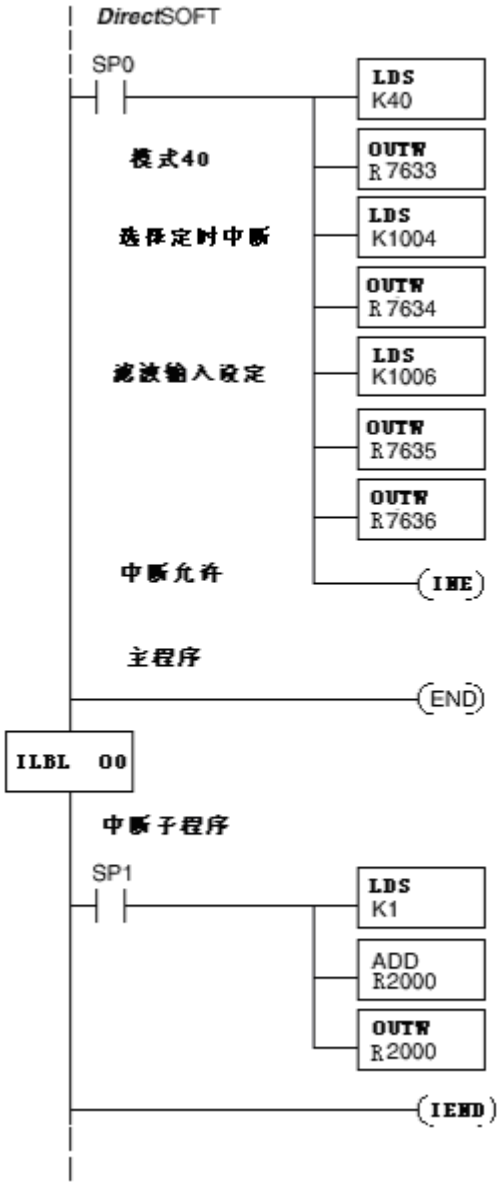
从中断程序返回

定时中断程序举例

下列程序选用了模式 **40**, 然后选择定时中断，定时中断周期为 **100mS**。



输入端 **I0**，**I1** 和 **I2** 被设置成滤波输入端，时间常数为 **10mS**。注意 **I0** 使用 **I1** 的时间常数。



K40 读入累加器，即选择模式 40 为 HSIO 方式

输出到 R7633（HSIO 方式选择寄存器地址）

选择中断时间常数 K1004，采用 100ms 周期

输出到 R7634，设置 100ms 周期中断时间

选择 10ms 滤波时间常数作滤波输入的常数

输出到 R7635，设置 I1 和 I0

输出到 R7636，设置 I2

允许中断

在这里插入主程序

主程序结束

中断服务子程序开始

K1 读入累加器

将累加器里的数与 R2000 中的值相加

结果输出到 R2000

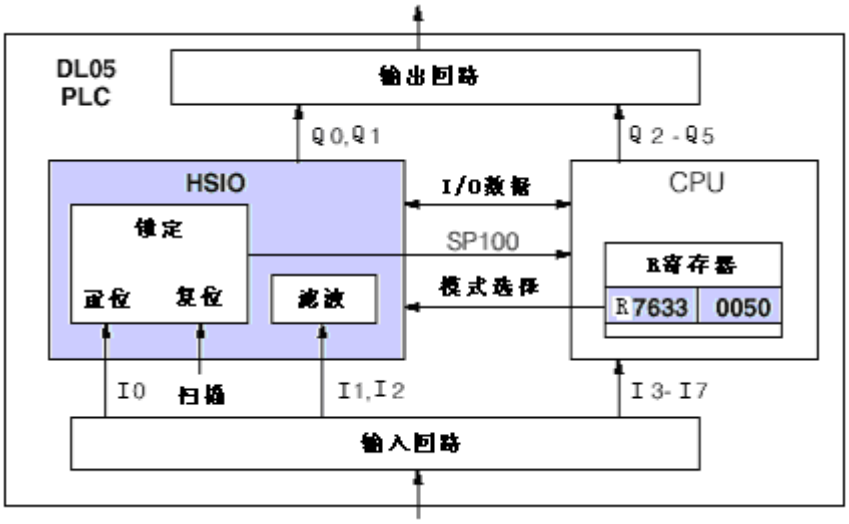
返回到主程序

第六节　模式 50：脉冲捕捉

HSIO 电路具有脉冲捕捉工作方式。它监控输入端 **I0** 上的信号，采集窄脉冲信号。脉冲捕捉方式的作用是使程序能“看见”脉冲宽度比当前扫描周期更短的输入信号。**HSIO** 电路锁定输入端 **I0** 的信号并保持一个扫描周期。在下一个扫描周期末尾，此触点自动断开。

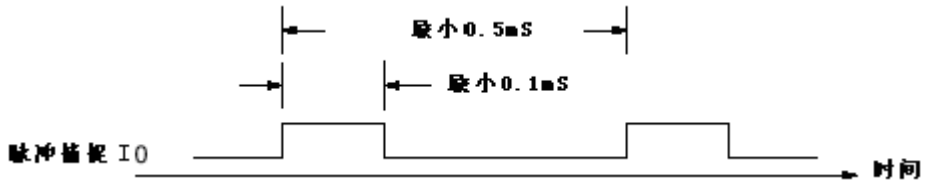
逻辑功能图

参阅下列逻辑图。当 **HSIO** 方式寄存器 **R7633** 的低字节是 **BCD** 码“**50**”时，**HSIO** 电路中的脉冲捕捉方式起作用。**I0** 自动变成脉冲捕捉输入端，它锁定每一个上升沿。在下一个扫描周期末尾，**HSIO** 复位锁定信号。输入端 **I1** 和 **I2** 用作滤波输入端。



脉冲捕捉时序

I0 上的脉冲信号必须符合某个定时标准，以保证捕捉到脉冲。参阅下面的时序图。**I0** 的输入特性是固定的（它不是一个可软件设定滤波时间的点）。最小脉宽是 **0.1mS**。在第二个脉冲到来之前，必须有一段延时，所以脉冲周期不能小于 **0.5mS**。如果脉冲周期小于 **0.5mS**，第二个脉冲将被看作是当前脉冲的一部分。



注意：脉冲捕捉和滤波输入功能在本质上是对立的。**I0** 端的脉冲捕捉的特点是为了要捕捉窄脉冲，而 **I1** 和 **I2** 端的滤波输入的特点是为了要去掉窄脉冲。

模式 50 设定

在 HSIO 模式选择寄存器 **R7633** 设置 BCD 码 50，选择**模式 50**。

R7633 = 0050 （BCD）

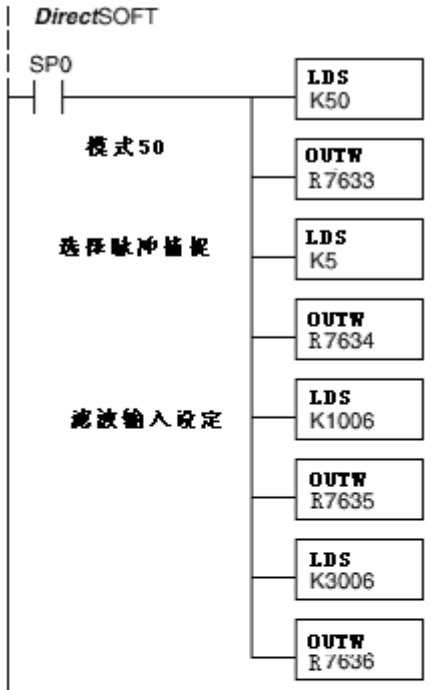
模式 50 下，I0~I2 的设定

下表列出了脉冲捕捉方式下 I0~I2 的输入选项设置。输入端 **I0** 为脉冲输入，并且配置寄存器 **R7634** 中内容必须固定为“**0005**”。输入端 **I1** 和 **I2** 只能是滤波输入。每个输入端都有各自的设置寄存器和滤波时间常数。

输入	配置寄存器	功能	需要的十六进制码
I0	R7634	脉冲捕捉输入	**0005 （固定）**
I1	R7635	滤波输入	××**06(**××=滤波时间**)** **0-99ms(BCD)**
I2	R7636	滤波输入	××**06(**××=滤波时间**)** **0-99ms(BCD)**

脉冲捕捉程序举例

下列程序选用**模式 50**, 然后用编程方法把 **I0** 设成脉冲捕捉方式。输入端 **I1** 和 **I2** 被设置成滤波输入端，时间常数分别为 **10 mS** 和 **30mS**。



K50 读入累加器，即选择模式 50 为 HSIO 方式

输出到 R7633（HSIO 方式选择寄存器地址）

将 I0 作脉冲捕捉输入

输到 R7634，设置 I0 为脉冲捕捉输入

选择 10ms 滤波时间常数的滤波输入

输出到 R7635，设置 I1

选择 30ms 滤波时间常数的滤波输入

输出到 R7636，设置 I2

使用脉冲捕捉输入将 Q0 置为 on，作用在 I0 上的是很短暂的脉冲

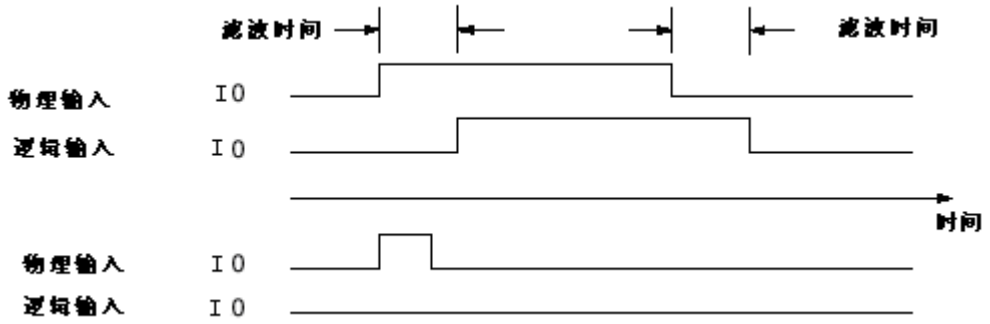
主程序结束

第七节　模式 60：滤波输入

我们将讨论 **HSIO** 电路的最后一种方式，就是**模式 60—软件**滤波输入。此方式的作用是让输入回路去掉窄脉冲，接收宽脉冲。它尤其适用于干扰环境下或其它脉冲宽度较重要的场合中。在这一章中提到的所有其它方式里，**I0-I2** 一般作为特殊输入端来支持其使用功能。只有不用的输入端才缺省作为滤波输入端用。现在，在模式 **60** 中，所有三个输入端 **I0-I2** 都只可作为软件滤波输入端用。

输入滤波时序

用一段延时时间来对输入端 **I0-I2** 上的信号脉冲进行滤波。在下图中，在时序线上面的物理输入信号的输入脉冲比滤波时间长，结果程序的逻辑输入是按滤波时间的上升沿和下降沿进行移相（延时）的。而在时序线下面的输入波形中，实际输入脉冲宽度比滤波时间短。在这种情况下，程序的逻辑输入保持在 **OFF** 状态（输入脉冲被滤掉了）。



模式 60 设定

在 HSIO 模式选择寄存器 **R7633 中**设置 BCD 码 60，选择**模式 60**。

R7633 = 0060 （BCD）

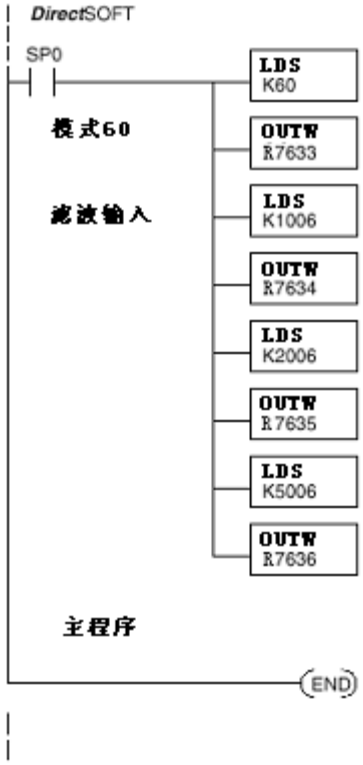
模式 60 下，I0~I2 的设定

下表列出了软件滤波输入方式下各滤波输入点的选项设置。滤波时间常数（延时）可设定为从 **0-99 mS**（当时间常数被设为 **0** 时，输入端作为普通输入端）。它的设置是以 **BCD** 码方式设定配置寄存器的高字节。我们把这个数码与低字节中的“ **06** ”合在一起，得到“ ××**06** “ ，此处××=**0-99**。输入端 **I0**，**I1** 和 **I2** 只能是滤波输入端。每个输入端都有各自的配置寄存器和滤波时间常数。

输入	配置寄存器	功能	需要的十六进制码
I0	R7634	滤波输入	××**06 (**××=滤波延时时间**)** **0-99ms(BCD)**
I1	R7635	滤波输入	××**06(**××=滤波延时时间**)** **0-99ms(BCD)**
I2	R7636	滤波输入	××**06(**××=滤波延时时间**)** **0-99ms(BCD)**

滤波输入程序实例

下列程序选择了模式 **60**，然后用编程方法设置了输入端 **I0**，**I1** 和 **I2** 的滤波延时时间常数。每个滤波常数各不相同，如图所示。



K60 读入累加器，即选择模式 60 为 HSIO 方式

输出到 R7633（HSIO 方式选择寄存器地址）

选择 10ms 滤波时间常数的滤波输入

输出到 R7634，设置 I0

选择 20ms 滤波时间常数的滤波输入

输出到 R7635，设置 I1

选择 50ms 滤波时间常数的滤波输入

输出到 R7636，设置 I2

插入主程序

主程序结束

第 5 章 系统维护

硬件系统的维护

维护标准： 该产品的维护并没有什么规则，然而，大约每隔一二个月，要对你的 PLC 和控制系统进行例行检查和维护，而且要包括以下几项：

- 空气温度——检查控制柜内空气温度，不要超出任何元件的操作温度范围。
- 空气过滤器——如果控制柜有空气过滤器，要定期清洗或更换。
- 保险丝或断路器——检验保险丝和断路器应完好。
- 清洁单元——检查所有的空气通风口是否干净。如果外箱体需要清洗，不要连接输入电源，用一块湿布仔细擦洗箱体。注意不要使水通过通风口进入箱体，不要用强清洁剂，因为这样可能损坏箱体油漆。

诊断

诊断： DL05 微型 PLC 在每个 CPU 扫描中进行预定义的例行诊断。该诊断能检测到 PLC 的变量错误或失败。两个主要的错误等级为致命的和非致命的。

致命错误： 致命的错误就是可能引起系统功能紊乱，也许可能引起安全问题的错误。如果此时 CPU 处于运行模式，CPU 将自动切至编程模式（注意：在编程模式所有输出被关闭）。当 CPU 在编程模式时，如果检测到致命的错误，CPU 将不允许你切至运行模式，除非此错误已解决。
下面是致命错误的几个例子：

- 电源供给失败。
- 奇偶错误或 CPU 故障。
- 特殊编程错误。

非致命错误： 非致命错误就是需要你的注意但不会引起操作紊乱的错误。这种错误不会引起也不会阻止 CPU 模式转换。应用程序能使用特殊继电器来测试非致命错误，如果需要的话，甚至使系统按顺序关闭或者使 CPU 切至编程模式。
下面是非致命错误的一个例子：

- 特殊编程错误。

查找诊断信息：当在线时，如果有一个错误发生，编程装置将会显示错误信息。

- DirectSOFT 显示这个错误号和一个错误信息。
- 手持编程器显示错误号和这个错误的短的描述信息。

错误码的位置：下表为对应一确定的错误信息类型的特殊存储器。

错 误 等 级	错 误 种 类	诊断存储器
用户自定义	使用了错误语句的错误代码	R7751
系统错误	致命的错误代码	R7755
	主要的错误代码	R7756
	一般的错误代码	R7757
语法错误	发生文法错误的地址	R7763
	在文法检查过程中发生的错误代码	R7764
CPU 扫描	CPU 从编程模式转换为运行模式后的扫描数	R7765
	当前扫描时间	R7775
	最小扫描时间	R7776
	最大扫描时间	R7777

DL05 微型 PLC 错误代码：

以下这些错误是 CPU 或手持编程器在不同的时间检测到的。然而，大多数是在刚接通电源、进入运行模式时、或者手持编程器键输入错误或发出一个非法申请时产生的。

错误代码	描 述
E003	软件暂停
E004	无效的语句（CPU 中的 RAM 奇偶错误）
E104	写失败
E151	无效的指令
E311	通讯错误 1
E312	通讯错误 2
E313	通讯错误 3
E316	通讯错误 6
E320	超时
E321	通讯错误
E360	手持编程器端口通讯超时
E501	非法进入
E502	非法地址
E503	非法指令
E504	非法参考/值
E505	无效的语句
E506	无效的操作
E520	非法操作——CPU 在运行模式
E521	非法操作——CPU 在测试运行模式
E523	非法操作——CPU 在测试编程模式
E524	非法操作——CPU 在编程模式
E525	模式开关不在编程位置
E526	离线单元
E527	在线单元
E528	CPU 模式
E540	CPU 锁定
E541	错误口令
E542	口令复位
E601	存储器满

E602	语句丢失
E604	参考信息丢失
E620	存储器溢出
E621	EEPROM 存储器不空
E622	无手持编程器 EEPROM
E624	仅仅 R 存储器
E625	仅仅编程
E627	非法写操作
E628	存储类型错误（应为 EEPROM）
E640	失去比较
E650	手持编程器系统错误
E651	手持编程器 ROM 错误
E652	手持编程器 RAM 错误

编程错误代码：

下表列举编程文法和运行时错误代码。从编程模式转换到运行模式或当你使用 AUX21 检查程序时会检测到的错误。CPU 将置位 SP52 并将错误代码存入 R7755。

错误代码	描　　述	错误代码	描　　述
E4**	CPU 中无程序	E438	无效的 IEND 地址
E401	丢失 END 状态	E440	无效的数据地址 DLBL
E402	丢失 GLBL、CLBL	E441	非 ACON/NCON 指令
E403	手持编程器端口通讯超时	E451	非法 MLS/MLR
E404	手持编程器端口通讯超时	E453	丢失 T/C
E405	手持编程器端口通讯超时	E454	非法 ATMR
E406	丢失 IEND	E455	非法 CNT
E412	GLBL/CLBL>64	E456	非法 SR
E421	级定义号重复	E461	堆栈上溢出
E422	GLBL/CLBL 定义号重复	E462	堆栈下溢出
E423	手持编程器端口超时	E463	逻辑错误
E431	无效的 ISG/SG 地址	E464	回路无连接
E433	无效的 CLBL 地址	E471	线圈定义号重复
E434	无效的 RET	E472	TMR 定义号重复
E435	无效的 CEND	E473	CNT 定义号重复
E436	无效的 ILBL 地址	E499	打印语句表
E437	无效的 RETI		

CPU 显示

DL05 微型 PLC 的前面板显示可以帮助你决定系统出现的问题。在通常的运行模式下，RUN 与 PWR 指示灯亮。可能的问题如下：

指 示 状 态	可 能 的 问 题
PWR（OFF）	系统电压不正确
	PLC 供给电源错误
RUN（OFF）	CPU 编程错误
	CPU 在编程模式
CPU（ON）	电气干扰问题
	内部 CPU 缺陷

通讯问题

如果你不能同 CPU 建立通讯，检查以下几项：

- 电缆没有连接。
- 电缆断线或连接不正确。
- 电缆不合适的端口或接地。
- 被连接的装置设定不正确的波特率（9600）。
- 被连接的装置的端口不正确。
- 两个装置的接地不同。
- 电气干扰引起通讯断断续续的错误。
- PLC 通讯端口坏并应更换。

在微机上用 DirectSOFT 软件的通讯问题，请看 DirectSOFT 手册。该手册包括通讯端口设置、地址或中断冲突等的诊断和问题的解决方法。

I/O 点问题的解决

可能的原因 I/O 错误的原因可能有以下几个方面引起：

- 高速 I/O 设置错误。
- 在你的机器或面板上的保险丝熔断（DL05 无内部 I/O 保险丝）。
- 端子块松动。
- 辅助 24 VDC 供给失败。
- 输入、输出电路失败。

快速解决问题的步骤

有助于你快速解决 DL05 微型 PLC 的 I/O 问题的一些事件如下:

- 高速 I/O 设置错误是 I/O 错误的普遍现象。如果发生问题的 I/O 点为 I0-I2 或 Q0-Q1，检查所有的参数设置表以适应你所选择的高速 I/O 模式。
- 输出电路不能短路或输出点开路。如果你怀疑一个或更多的错误点，要检测从公共点到怀疑点间的电压。注意当使用数字电压表时，必须注意从诸如可控硅管或晶体管类输出装置的漏电流。

- PLC 上的 I /O 状态指示灯是表示逻辑端的状态的。对于输入，其状态灯亮表示输入点正常，请确认当断开输入点连线时，其对应的状态指示灯是否熄灭；对于输出，其状态灯亮，并不表示输出点一定正常。
- 当连接现场设备到 PLC I/O 点时，漏电流可能是问题源。当某一输出设备所产生的漏电流大得足以使所连接的输入设备导通时，则会产生误输入信号。为避免这种情况，可在输入或输出上并上一个电阻。阻值得大小根据漏电流以及当时的电压而定。通常情况下采用 10K~20KΩ的电阻。请确认电阻的功率符合你的应用。
- 由于 DL05 的接线端子设置成整体可插拔的，因此，在你有空余的 DL05 时，可采用整体更换 PLC 的办法来解决问题。但是，当你认为某个现场设备有问题时，你必须首先检查现场设备的情况。否则，即使你换了 PLC，还是会产生同样的问题。

电气干扰问题

干扰是最难处理的问题之一，电气干扰可以以多种方式进入系统，它们主要分成 2 大类：传导型和放射型。很难判断电气干扰是如何进入系统的，但对这 2 种干扰的纠正措施是相似的。

- 传导型干扰是通过连接电线，面板等电气连接传入系统的，例如：I/O 连接回路、工作电源线、通讯接地线、底板接地线等等。
- 放射型干扰是一种类似电波的干扰，无须通过电气连接就能进入系统。

电气干扰不能完全根除，但可以减小到不影响系统的程度。遵循以下几点可有效减少电气干扰。

- 大多数干扰是由于不正确的接地系统而引起的。一个好的接地系统可有效减小干扰问题，要确保所有接地线为单点接地，并且相互间不形成菊花链。
- 电气干扰可通过 PLC 或 I/O 回路的供电部分侵入系统。在所有 AC 供电回路中加接隔离变压器可有效解决此问题。DC 电源必须为有良好接地系统的高质量电源系统。
- 把输入连线和输出连线隔离，低压信号线决不与高压信号线混布。

程序调试

DL05 提供了几个方法，可帮助用户在进行实际的机器运行前，检测你的程序，包括：

- 程序文法检查；
- 定义号重复使用检查；
- 特殊指令的使用；
- 运行中修改程序；
- I/O 强制操作。

附录

附录一　DL05 指令表

顺序控制指令

分　　类		指　　令	允许操作数	备注
接点指令	逻辑运算开始 NO 接点	LD	I、Q、M、T、C、S、SP	
	逻辑运算开始 NC 接点	LDN	I、Q、M、T、C、S、SP	
	逻辑与运算 NO 接点	AND	I、Q、M、T、C、S、SP	
	逻辑与运算 NC 接点	ANDN	I、Q、M、T、C、S、SP	
	逻辑或运算 NO 接点	OR	I、Q、M、T、C、S、SP	
	逻辑或运算 NC 接点	ORN	I、Q、M、T、C、S、SP	
直接输入	逻辑运算开始 NO 接点	LDDI	I	
	逻辑运算开始 NC 接点	LDNDI	I	
	逻辑与运算 NO 接点	ANDDI	I	
	逻辑与运算 NC 接点	ANDNDI	I	
	逻辑或运算 NO 接点	ORDI	I	
	逻辑或运算 NC 接点	ORNDI	I	
微分接点	逻辑运算开始上升沿接点	LDPD	I、Q、M、T、C、S、SP	
	逻辑运算开始下降沿接点	LDND	I、Q、M、T、C、S、SP	
	逻辑与运算上升沿接点	ANDPD	I、Q、M、T、C、S、SP	
	逻辑与运算下降沿接点	ANDND	I、Q、M、T、C、S、SP	
	逻辑或运算上升沿接点	ORPD	I、Q、M、T、C、S、SP	
	逻辑或运算下降沿接点	ORND	I、Q、M、T、C、S、SP	
带设定值的 T/C 接点	逻辑运算开始 NO 接点	LD	1st: T、C　2nd: R、P、K	
	逻辑运算开始 NC 接点	LDN	1st: T、C　2nd: R、P、K	
	逻辑与运算 NO 接点	AND	1st: T、C　2nd: R、P、K	
	逻辑与运算 NC 接点	ANDN	1st: T、C　2nd: R、P、K	
	逻辑或运算 NO 接点	OR	1st: T、C　2nd: R、P、K	
	逻辑或运算 NC 接点	ORN	1st: T、C　2nd: R、P、K	
比较等于接点	逻辑运算开始等于接点	LDEQ	1st: R、P　2nd: R、P、K	
	逻辑运算开始不等于接点	LDNEQ	1st: R、P　2nd: R、P、K	
	逻辑与运算等于接点	ANDEQ	1st: R、P　2nd: R、P、K	
	逻辑与运算不等于接点	ANDNEQ	1st: R、P　2nd: R、P、K	
	逻辑或运算等于接点	OREQ	1st: R、P　2nd: R、P、K	
	逻辑或运算不等于接点	ORNEQ	1st: R、P　2nd: R、P、K	
比较大于等于接点	逻辑运算开始大于等于 NO 接点	LDGE	1st: R、P　2nd: R、P、K	
	逻辑运算开始大于等于 NC 接点	LDNGE	1st: R、P　2nd: R、P、K	
	逻辑与运算大于等于 NO 接点	ANDGE	1st: R、P　2nd: R、P、K	
	逻辑与运算大于等于 NC 接点	ANDNGE	1st: R、P　2nd: R、P、K	
	逻辑或运算大于等于 NO 接点	ORGE	1st: R、P　2nd: R、P、K	
	逻辑或运算大于等于 NC 接点	ORNGE	1st: R、P　2nd: R、P、K	

分类		指令	允许操作数	备注
逻辑组连接	逻辑组串联	ANDLD	----------	
	逻辑组并联	ORLD	----------	
母线指令	新母线开始	MLS	K (BCD)	
	母线复归	MLR	K (BCD)	
输出指令	线圈输出	OUT	I，Q，M	
	线圈输出(后优先)	ZOUT	I，Q，M	
	线圈置位	SET	I，Q，M，S	
	线圈复位	RST	I，Q，M，S	
	输出暂停	PAUSE	Q	
	直接输出	OUTDI	Q	
	直接输出(后优先)	ZDI	Q	
	线圈直接置位	SETDI	Q	
	线圈直接复位	RSTDI	Q	
微分	1 次扫描输出	PD	I，Q，M	
移位寄存器	移位寄存器	SR	M	
定时器计数器指令	0.1s 定时器	TMR	1st: T 2nd: R,P,K	
	0.01s 定时器	HTMR	1st: T 2nd: R,P,K	
	0.1s 累积定时器	ATMR	1st: T 2nd: R,P,K	
	0.01s 累积定时器	AHTMR	1st: T 2nd: R,P,K	
	计数器(带复位端)	CNT	1st: C 2nd: R,P,K	
	计数器(不带复位端)	GCNT	1st: C 2nd: R,P,K	
	加减计数器	UDCNT	1st: C 2nd: R,P,K	
	定时器/计数器复位	RSTTC	T，C	

程序执行控制指令

分类		指令	允许操作数	备注
级式指令	级登记	SG	S	
	初始级登记	ISG	S	
	条件成立时级转移	JMP	S	
	条件不成立时级转移	NJMP	S	
	级合流登记	CV	S	
	级合流转移	CVJMP	S	
循环	程序循环	FOR	R，K	
	循环结束	NEXT	----------	
子程序	调用	CAL	K	
	子程序	CLBL	K	
	子程序中断	RET	----------	
	子程序结束	CEND	----------	
中断	中断禁止	INH	----------	
	中断许可	INE	----------	
	中断程序标号	ILBL	O	
	中断程序条件返回	RETI	----------	
	中断程序结束	IEND	----------	
其他	WDOG 复位	WDOGR	----------	
	STOP 停止方式	STOP	----------	
	空操作	NOP	----------	
	程序结束	END	----------	

数据处理指令

分类			指令	允许操作数	备注
读入指令	16 位		LDW	R，P	
	32 位		LDD	R，P	
	任意位		LDF	1st: I,Q,M,S,T,C,SP 2nd: K	
	数据堆栈弹出		POP	----------	
	4 位常数		LDS	K	
	8 位常数		LDC	K	
	寄存器号		LDR	O	
写入指令	16 位		OUTW	R，P	
	32 位		OUTD	R，P	
	任意位		OUTF	1st: I,Q,M 2nd: K	
BCD 运算	加法	4 位	ADD	R，P	
		8 位	ADDD	R，P	
		8 位常数	ADDC	K	
	减法	4 位	SUB	R，P	
		8 位	SUBD	R，P	
		8 位常数	SUBC	K	
	乘法	4 位	MUL	R，P	
		4 位常数	MULS	K	
		8 位常数	MULD	K	
	除法	4 位	DIV	R，P	
		4 位常数	DIVS	K	
		8 位常数	DIVD	K	

分类			指令	允许操作数	备注
逻辑指令	与	16 位	ANDW	R，P	
		32 位	ANDD	R，P	
		8 位常数	ANDC	K	
	或	16 位	ORW	R，P	
		32 位	ORD	R，P	
		8 位常数	ORC	K	
	异或	16 位	XORW	R，P	
		32 位	XORD	R，P	
		8 位常数	XORC	K	
比较	16 位		CMPR	R，P	
	32 位		CMPRD	R，P	
	8 位常数		CMPRC	K	
ACC 变换	取和		SUM	----------	
	取反		INV	----------	
	BIN 变换		BIN	----------	
	BCD 变换		BCD	----------	
	格雷码变换		GRAY	----------	
	编码		ENCO	----------	
	译码		DECO	----------	
	位替换指令		SFLDGT	----------	
	右移		SHFR	R，K	
	左移		SHFL	R，K	
寄存器变换	BCD 增 1		INCR	R，P	
	BCD 减 1		DECR	R，P	
	BIN 增 1		BINC	R，P	
	BIN 减 1		BDEC	R，P	
数据块处理	传送		MOVE	R，P	
	ASCII→HEX 变换		ATH	R	
	HEX→ASCII 变换		HTA	R	
	数据标号		DLBL	K	
	数值登记		NCON	K	
	ASCII 登记		ACON	A	
	数据标号地址		LDLBL	K	
	存储器传送		MOVMC	R，K	
	ASCII 码打印		PRINT	ASCII	

分类			指令	允许操作数	备注
BIN运算	加法	4位	BADD	R，P	
		4位常数	BADDS	K	
	减法	4位	BSUB	R，P	
		4位常数	BSUBS	K	
	乘法	4位	BMUL	R，P	
		4位常数	BMULS	K	
	除法	4位	BDIV	R，P	
		4位常数	BDIVS	K	

模块对象特殊指令

分类		指令	允许操作数	备注
通讯	读	RX	I,Q,M,T,C,SP,S,$,R,P	
	写	WX	I,Q,M,T,C,SP,S,$,R,P	
日历	时间设定	TIME	R	*
	日期设定	DATE	R	*
其他	输出暂停	PAUSE	Q	
	诊断代码显示	FALT	R,K	

*注：DL05 的日历时钟指令仅在使用 D0-01MC 时有效。

附录二　DL05 系列无协议通讯

DL05 系列通用通讯口 2 支持无协议通讯（V3.0 以下），其通讯参数的设置是通过特殊寄存器设定的方式进行的，能设定的各可能参数如下表：

设定项目	初始设定值	可设定范围
协议	无协议通讯	－－－－－－
通讯方式	X-ON/X-OFF 无效 RTS 信号控制无效	X-ON/X-OFF　无效/有效 RTS 信号控制　无效/有效
通讯速度	19200bps	300/600/1200/2400/4800/9600/19200/38400bps
数据位长	8 bit	7bit/8bit
START 位数	1bit	固定
STOP 位数	1bit	1bit/2bit
奇偶校验	奇校验	奇数（ODD）/偶数（EVEN）/无（NONE）
受信结束时间设定	3 字符时间	3 字符时间+0/2/5/10/20/50/100/500ms
RTS ON 延时	0ms	0/2/5/10/20/50/100/500ms
RTS OFF 延时	0ms	0/2/5/10/20/50/100/500ms

相关的特殊寄存器列表如下

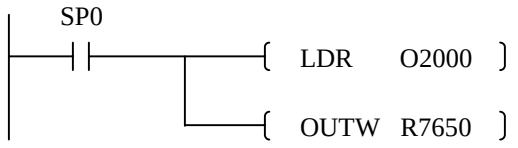
寄存器号	功能	内容	备注
R7650	接收数据存放寄存器号	寄存器号（8 进数），初始值：0H	
R7651	----------	----------	
R7652	----------	----------	
R7653	通讯结束码	接收结束码 例如：CR（0DH）	
R7654	----------	----------	
R7655	通讯协议选择、延时时间	K/N/MODBUS/M-NET/无，延时时间	
R7656	通讯参数	通讯方式、速度、数据位、停止位、级；奇偶位	
R7657	R7655/7656 设定结束确认	0A00/0E00	

各设定寄存器的具体说明如下：

1） R7650 接收数据存放寄存器号的设定

设定无协议通讯时存放通讯接收数据的寄存器区域的首址。可设定的寄存器号，原则上为数据寄存器区域，作为首址的可设定范围为 R1400（0300h）~R7000（0E00h），R7650 中的初始值为 0h。

设置 R2000 为通讯接收区域首址的程序例子如下：



通讯接收数据序列的格式如下：

设定寄存器	高字节为 00，低字节为通讯状态字(STS)
	STS = 80h：正常通讯结束
	C0h：通讯出错
	F0h：接收数据溢出(>128 字节)
设定寄存器 + 1	接收数据字节数（BIN 码），最大 128 字节（80h）
设定寄存器 + 2	从该寄存器的低字节开始存放通讯接收数据。

当设定的寄存器号在允许范围外的时候，系统自动把接收数据寄存器号设定为 R7000。

假设，通讯接收寄存器为 R2000，接收到下列数据时的寄存器情况如下：

第 35 字节		5	4	3	2	第 1 字节
EF	→	9A	78	56	34	12

数据接收、寄存器存放的顺序：

寄存器号	高字节	低字节
R2000	00	STS
R2001	00	23h
R2002	34	12
・・・・・・・・・・・・・・・・・・		
R2023	**	EF

当 DL05 判断通讯数据接收结束时，它会把接收到的数据顺次存放在设定寄存器+2 开始的寄存器组中，如上例所示。

DL05 一次通讯最多接收 128 个字节，当接收数据大于 128 字节时，从 129 字节开始的数据将被忽略，接下来，在通讯数据接收结束点，在设定寄存器中设置 STS = F0h，以表示数据接收溢出，同时，把前 128 个接收数据存放到接收数据寄存器中。

2） R7653 通讯结束码设定

在 DL05 无协议通讯中，需要设置通讯结束码用于表示通讯数据接收结束。你可选择 1 字节结束码，2 字节结束码或不设置结束码，初始值为无结束码(00h)。

	高字节	低字节
R7653	结束码 2	结束码 1

通讯结束码可有 3 种选择

（1） 1 字节通讯结束码

结束码 1 为通讯结束码，结束码 2 = 00h

例，当要把 CR（0Dh）作为通讯结束码时，设定

R7653 = 000D

（2） 2 字节通讯结束码

在 R7653 中设置 2 字节通讯结束码。

例，当要把 CR（0Dh），LF（0Ah）作为通讯结束码时，设定

R7653 = 0A0D

（3） 无通讯结束码

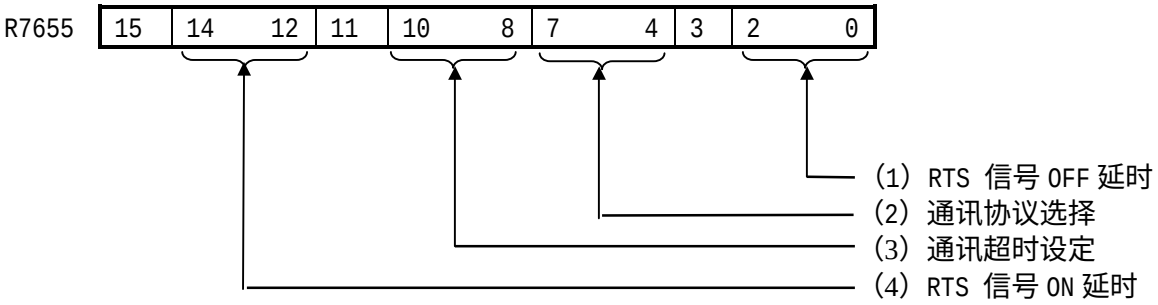
此时，结束码 1 = 00h；结束码 2 = 00h。DL05 根据受信结束时间来判定通讯接收结束。

注意：当结束码 1 = 00h 时，即使结束码 2 不为 0，也作为无通讯结束码处理。

当往设定结束确认寄存器 R7657 中写入“ 0500 ”时，如果在通讯结束码设定中有错误，则系统会在 R7657 中写入一个“ 0E00 ”。

3） R7655 通讯选择，通讯延时时间设定寄存器

R7655 组成如下：



(1) R7655（Bit2—0）RTS 信号 OFF 延时设定

数据发送结束后，RTS 信号的 OFF 延时时间的设定码。

000（0） = 无（0ms）	100（4） = 20ms
001（1） = 2ms	101（5） = 50ms
010（2） = 5ms	110（6） = 100ms
011（3） = 10ms	111（7） = 500ms

(2) R7655（Bit7—4）通讯协议选择设定

Bit7 为 1：(8*h) = K 协议选择

Bit6 为 1：(4*h) = CCM 协议选择

Bit5 为 1：(2*h) = MODBUS 协议选择

Bit4 为 1：(1*h) = 无协议通讯选择

(3) R7655（Bit10—8）通讯超时设定

在数据接收方式下，可设定以下的字符等待时间，用于判断数据接收通讯的结束。当接收一个字符后，经过了以下的设定时间还接收不到下一个字符，系统就判断本次数据接收通讯结束，把此前接收到的数据送入数据接收寄存器中，结束本次接收通讯。

可设定的时间值如下：

000（0） = 无（0ms）	100（4） = 20ms
001（1） = 2ms	101（5） = 50ms
010（2） = 5ms	110（6） = 100ms
011（3） = 10ms	111（7） = 500ms

(4) R7655（Bit14—12）RTS 信号 ON 延时设定

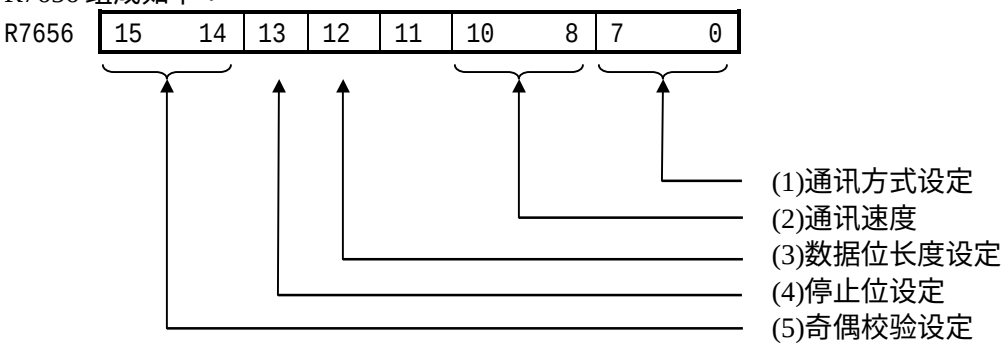
RTS 信号 ON 后，开始发送数据延时时间的设定

000（0） = 无（0ms）	100（4） = 20ms
001（1） = 2ms	101（5） = 50ms
010（2） = 5ms	110（6） = 100ms
011（3） = 10ms	111（7） = 500ms

当选择 RTS 控制无效（70h，71h）的情况下，本设定数据无效。

4） R7656 通讯参数设置寄存器

R7656 组成如下：



（1） R7656（Bit6—0）通讯方式设定

在这儿设置无协议通讯的方式。在 DL05 的无协议通讯中，可选择 RTS 控制有效/无效，X-ON/OFF 有效/无效。具体设定如下：

无协议通讯方式码	通讯方式	通讯流程控制	
		X-ON/X-OFF	RTS 信号
70h	可变格式 A（串行接收发送）	无效	无效
71h	可变格式 B（串行接收发送）	有效	无效
72h	可变格式 C（串行接收发送）	无效	有效
73h	可变格式 D（串行接收发送）	有效	有效

（2） R7656（Bit10—8）通讯速度

通讯速度可在以下 8 种中自由选择，记住，通讯双方的通讯速度必须一致。

000（0） = 300bps　　100（4） = 4800bps
001（1） = 600bps　　101（5） = 9600bps
010（2） = 1200bps　　110（6） = 19200bps
011（3） = 2400bps　　111（7） = 38400bps

（3） R7656（Bit12）数据位长度设定

指定通讯数据的字节位长。

0 = 8 Bit　　1 = 7 Bit

（4） R7656（Bit13）停止位设定

指定通讯数据的停止位数。

0 = 1 Bit　　1 = 2 Bit

（5） R7656（Bit15—14）奇偶校验设定

00、01 = 无校验（NONE）　10 = 奇校验（ODD）　11 = 偶校验（EVEN）

5) R7657 通讯设定结束确认寄存器

当以上的通讯设定寄存器设置完了以后，在 R7657 中写入“ 0500 ”，以告诉系统有新的通讯参数设定。

设定 OK，系统置 R7657 =“ 0A00 ”；设定 NG，系统置 R7657 =“ 0600 ”或“ 0E00 ”。（“ 0600 ”表示局号指定错误；“ 0E00 ”表示通讯结束码指定错误。）

6) 无协议通讯特殊寄存器设定例子

下面的设定例子中，设定通讯口 2 为无协议通讯，其接收数据存放开始地址为 R2000，接收通讯结束码为 CR（0Dh），通讯控制 X-ON/XOFF，RTS 信号为无效（模式 70h），接收通讯超时等待时间为 100ms，通讯数据为 1 位停止位，8 位数据位，无校验通讯，通讯速率为 9600bps，程序如下：

```
   SP0
|---| |----[ LDR   O2000]    设置通讯接收寄存器开始地址为 R2000
|       |----[OUTW   R7650]
|       |----[LDS    K000D]    通讯接收结束码为 CR（0Dh）
|       |----[OUTW   R7653]
|       |----[LDS    K0610]    设定超时等待时间 100ms，选择无协议通讯
|       |----[OUTW   R7655]
|       |----[LDS    K0570]    无协议通讯参数设定
|       |----[OUTW   R7656]
|       |----[LDS    K0500]    通讯设定结束码设定
|       |----[OUTW   R7657]
```

上面的设置程序运行后，则通讯口 2 处于无协议通讯等待状态，你可直接向 DL05 发送数据，DL05 把所接收到的数据按规定的格式存放于接收寄存器中。当要在无协议通讯下发送数据时，可编制如下的数据发送程序。

```
  条件   SP116                          通讯发送开始条件
|-----| |-------|/|----------[LDS   KF202]   F2 固定，02 为通讯对方局号（无协议通讯下任意）
|                 |----------[LDS   K4]      通讯字节数（最大 128 字节）
|                 |----------[LDR   O2500]   发送数据存放地址
|                 |----------[WX    R0]      数据发送指令，在无协议通讯下，R0 可为任意合法地址
```

以上程序为与计算机进行无协议通讯的一个例子，假设 R2500，R2501 中存放有以下数据，R2500 = K3231，R2501 = K3433，则执行一次发送指令，利用计算机上的超级终端，可接收到 1234 的数据（先接收到 1，再接收到 2，然后为 3，最后为 4）。

光洋电子(无锡)有限公司
Koyo ELECTRONICS（WUXI） CO., LTD.
地址:江苏省无锡市蠡溪路 118 号 邮编:214072
电话：0510-5167888 传真：0510-5161393
http://www.koyoele.com.cn

KEW-M4111A

2004 年 2 月